PERSONALNI
računari
BIGZ
89
FEBRUAR
1993.
CENA 20.000 D
SLT 500 ○ Denara 16000 ○ DEM 10 ○ USD 6 ○ ATS 70

SOFTVER

WINDOWS FOR WORKGROUPS
AMI PRO FOR WINDOWS
QUATTRO PRO FOR WINDOWS
PKZIP 204C

SPECIJALNI DODATAK NA 32 STRANE

MICROSOFT WORD 2.0

REFERENTNI PODSETNIK

PKZIP 204c

UPOREDNI TEST

10 SVGA MONITORA

INTERVJU

DR MILAN BOŽIĆ

MIKROPROCESORI

KAKO IZABRATI
PRAVI PROCESOR?

LIDVERI

COMPAQ SYSTEMPRO

CAD RAČUNARI

CAD RAČUNARI I OPREMA

CAD računari i oprema
386 i 486 računari optimizovani za CAD i grafičke namene
DIAMOND i HERCULES procesorske grafičke karte
EIZO kolor monitori visoke rezolucije
HEWLETT-PACKARD kolor štampači, skeneri i ploteri
POLAROID film rekorderi

SOFTWARE

AutoCAD r.12
AutoSHADE
ANiMATOR Pro
3D Studio 2

USLUGE

AutoCAD škola
CorelDRAW škola
RENTIRANJE plotera, laserskih i kolor štampača
kolor skenera,
film rekordera

KONSALTING

ArchiCAD i Archi 24
globalna rešenja za primenu CAD-a u arhitekturi

OUT PRESS

CAD RAČUNARI STVARAJU UMETNOST

PREDUZECE
MICRO ANIMA
Beograd, 29 novembra 71
Tel 343 940 / 343 294

MICRO

ANIMA

računari 89

časopis za informatiku i računarstvo
YU ISSN 0352-7221
Izdaje BIGZ
februar 1993

Izdaje i štampa
Beogradski izdavačko-grafički zavod
11000 Beograd
Bulevar Vojvode Mišića 17
Generalni direktor
Ilija Rapajić
Glavni i odgovorni urednik
Jovan Regasek
Komercijalni urednik
Vesna Jeremić
Prevodilac
Ranka Jovanović
Tehnički urednik
Radoje Kavedžić
Ilustratori
Dragan Kovačević
Dejan Medić
Stručna redakcija
Nenad Bašošanin (baze podataka)
Ranka Jovanović (aktuelnosti) Dejan Ristanović (softver) Jovan Skuljan (programiranje) Vladimir Stamenović (radne tabele) Dejan Veselinović (hardver) Dragutin Vuković (mreže), Zoran Životić (softver)
Adresa redakcije
11000 Beograd
Bulevar vojvode Mišića 17/III
Telefoni
653-748 (redakcija), (prodaja) 651-666 i 258
Agencija BIGZ-a
(011) 651 793 653-585
Telefaks
(011) 648-140 647-955
Pretplata za zemlju
Za šest meseci zvati prodaju
Za jednu godinu zvati prodaju
(na žiro-račun D P BIGZ 60802-603 23264)
Pretplata za inostranstvo
Za jednu godinu $1 USD, 80 DEM 67 CHF 27 GBP 271 FRF
(na devizni račun D P BIGZ 60811-620-16101-820701-999-03377)
Zbog usporenog prenošenja uplate preko banke molimo pretplatnike da nam posle svake nove uplate odmah pošalju foto-kopiju uplatnice
Rukopisi se ne vraćaju

SEZAM (011) 648-899 (deset linija)
sistem za modemske komunikacije
Upravnik sistema
Jovan Regasek
Administratori na sistemu
Zoran Životić i Dejan Ristanović

CENOVNIK OGLASNOG PROSTORA

Cene oglasnog prostora se izražavaju u „bodovima" čija vrednost odgovara vrednosti nemačke marke na slobodnom tržištu na dan fakturisanja.

• 1/1 poslednja strana korica	1 000 bodova
• 1/1 druga strana korica	900 bodova
• 1/1 treća strana korica	900 bodova
• 1/1 kolor unutrašnja strana	900 bodova
• 1/2 kolor unutrašnja strana	900 bodova
• 1/1 crno-bela strana	650 bodova
• 1/2 crno-bela strana	900 bodova
• 1/4 crno-bela strana	380 bodova
• 1/8 crno-bela strana	900 bodova
• mali oglas	50 bodova

Fakturisanje se vrši na dan ogovaranja oglasnog prostora i prijema potrebnih materijala za oglas sa obavezom da se uplata izvrši pre izlaska broja iz štampe

Rok za slogovar i rezervaciju oglasnog prostora je 35 dana pre izlaska broja iz štampe (25. oktobar za decembarski broj)

Molimo vas da se za slogovare i informacije javite na telefon 011/653-748 fax 011/648-140 modem 011/648-899 (mail write redakcija) Kontakt: Vesna Jeremić

SADRŽAJ

4 Vesti
ŠTA IMA NOVO
10 Redne stanice
ILI-, SINHRAN, KOMPAKTAN
13 Grafičke kartice / Superstation 3D
SVA SILA DAJE UBRZANJE
15 Polamike / Local bus
OBRAČUN KOD O.K. LOKALA
18 Moniton / Uporedni test
NAJPLOŽIJA BOJA
22 Serveri / Compaq SystemPro
SNAGA PO MERI
26 Tehnologija / Mikroprocesori
POVUCITE PRAVI POTEZ
32 O čemu misli, šta radi / Dr Milan Božić
MOJ VIRTUELNI SVET
36 Radne tabele / QPW
QUATTRO... PA OSTALI
40 Obrada teksta / AmiPro 3.0
NAJBOLJI PISEV PRIJATELJ
46 Operativni sistemi /
Windows for Workgroups
USLUGA ZA USLUGU
52 Arhivan / PKZIP 204c
ZIP JE STIGAO
56 Prvi utisci / ACROS 325SE
BELEŽNICA
60 Kompajlan / Fortran PlusFort
PRVA LJUBAV, DRUGI PUT
62 Alatke / Stackar
ISTROŠENI DISKOVI
67 Projekti / Programski jezik Manevro
MOĆ REČI
70 Tehnike programiranja
OBUJMI U PAMĆENJU
74 Mikroprocesori / i486
ARHIVSKI PODACI
78 Algoritmi / Pretraživanje
LOGARITAMSKE METODE
82 Tehnike programiranja / Baza podataka
MALINA U KOJOJ SVE PIŠE
88 Tehnike programiranja / C++
KONTROLA VASTANJE
92 Dejanove pitalice
DELJIVO SA SEDAM
93 Zoran Životić
MOĆNI SINGLA C-a
94 Dejan Ristanović
MATEMATIČKE PRIRODE
96 Bilteni
SEZAMI BILTEN
98 Polamika
PRESTUP S PREDUMIŠLJAJEM

Na naslovnoj strani: Računari su postali već toliko uobičajena stvar da ih koriste svi, pa čak i (barem prema zamisli našeg fotografa Ljubiše Tešića) - anđeli na nebu. Pošto ove neobične korisnike, iz razumljivih razloga, nismo mogli da snimimo u prirodnom ambijentu, zamolili smo sekretericu „Sezama" Aleksandru Jovanović da nam pomogne da dočaramo nebesku atmosferu. Snimak: Studio Tešić & Nenad Petrović

SADRŽAJ OGLASA

AB SOFT	76
ABACOM	44,47
ABA COMPUTER	73
ASTS COMMERCE	35
BEST	43
CET	12,20
CONTEK	30
CONTRAD	14,48
DIAGRA	35
DITEH	9
DEC COMMERC	66
DATEK	7,45
DUNO-D	35
INSTITUT „VINČA"	96,97
INTERSOFT	29
JUGOBATA	11
MICRO AIR	28
MICRO KNJIGA	25
MICROSYS	20,21
MIKROSERVA	26
MIMACO	81
MP DUNO	77
MR SYSTEM	65
OLYMP	73
OMA	5
PTC	57
PERUNAC	54
PREDUZEĆE MR	25
PROSOFT	55
PS CLUB	51
PS PLUS	39
SAGA	49,75
SOFT PROJEKT	39
SOFTEK	64
TRIMCOM	65

OPERATIVNI SISTEMI

OS/2

Beta-verzija 2.1 odlazi na 5000 adresa

„IBM Personal Software Products" najavljuje betaverziju 32 bitnog operativnog sistema – OS/2 2.1. Betaverzija će otići ne više od 5000 adresa nezavisnim proizvođačima softvera i hardvera. OS/2 2.1 dozvoljava da se OS/2 *Windows 3.1* i DOS aplikacije izvršavaju istovremeno na istom displeju i ima novu 32 bitnu grafičku mašinu visokih performansi. Ugrađeni su i softverski drajveri za 256-kolor XGA i SVGA za brze štampače i podršku CD-ROM kao i mini-aplikacija za slanje i primanje faksiminih poruka. To još nije sve – tu je i kod za pen-računare za sistemske kojli primaju glasovna naređenja i za multimedia ekstenzije.

OS/2 2.1 podržava APM (Advanced Power Management – usavršeno upravljanje strujom) što produžava vek trajanja baterije kod *notebook* računara i hardverskih uređaja dimenzije kreditne kartice sa novom PCMCIA (*PC Memory Card International Association*) specifikacijom.

OS/2

OS/2 2.0 na kuri za mršavljenje

IBM će pokušati da ovoji novu korisničke novom verzijom svog operativnog sistema OS/2 koji će biti štedljiviji u korišćenju memorije. Tokom prve plovne ove godine OS/2 bi trebao da se vrati se kura za mršavljenja na koju su ga poslali IBM-ovi softverski inženjeri. Oni su putem ankete zakjučili da mnogim korisnicima i sistemskim menadžerima smeta baš tej preteranu obim u koma se OS/2 širi preko svih sistemskih resursa. Ako neko želi da na svom računaru pod OS/2 koristi mogućnosti višeprogramskog rada i da bez problema uporedo izvršava više aplikacija bolje da ni na početnje bez bar 8 MB radne memorije. A pravi višeprogramski rad je kako je nedavno pokazala anketa koju je sprovelo „Computerworld" za najveći broj korisnika glavna prednost OS/2 nad *Windows*-om. Ukoliko naka od aplikacija koja se izvršavaju krahira, ostatak sistema će neometano raditi dalje.

Halapljivo trošenje RAM memorija i prostora na hard-disku bili su glavni razlog da IBM poštalja OS/2 na „poeni tratman" Novi OS/2 koji će iz toga proizaći trebalo bi da buda tako vitak i „stesan" da može da se instalira i na *laptop* i *notebook* računara koji ratko mogu da se pročita na više od 4 MB radne memorije kao i ne jeftinije stona računara a da ostavi i prostora za pokretanje aplikacija.

Ipak se operativnim sistemom nija ista situacija kao npr se puterom bez masti kafom bez kolaina, ili duvanom bez nikotina. Kod operativnog sistema se ne žalost ne može štedeti ne isti način. Bez upravljanja memorijom i procesima i bez savremanog sistema datoteka, funkcionisa jedino DOS. Gde onda uštedeti? Alatkama i pomoćnim programima ovaj 32-bitni sistem je u poređenju sa *Windows*-om, iznako oskudno opremljen.

Iz IBM-a stižu uveravanja da u *Lita* verziji neće nedostajati nijedna od značajnijih mogućnosti sistema. Štaviše se isključivo na „katalo2enom salu" veliki dalovi koda biće prerađeni i bolje napisani, a suvišne stvari uklonjena.

To ne samo da zvuči dobro već i obećava da će ovaj operativni sistem raditi brže. Ali sistem neće biti samo iznutra pokretljiviji. Povećanjem slobodne radne memorije skratiće se vreme pristupa hard-disku kao i vreme potrebno za izvršavanje programa. U slučaju da tima ne buda umanjena funkcionalnost, korisnici će u *Lite* verziji dobiti ne jeđan celobijen operativni sistem već „sportsku" i bržu verziju starog sistema.

Da li je ovo suviše lepo da bi bilo istinito? Iza svega se verovatno krije ono što su u IBM u imali na umu pri novim preradama tiho tanelrak sledeću verziju OS/2 se novim imidžom. Za razliku od prethodnih godina kada su nove verzije OS/2 kasnile ili bile nedorađene ovaj put se čini da je „Veliki Plavi" ozbiljniji.

Glavni konkurent OS/2 sistemu – *Windows 3.x* – nalazi se na sva više računara made nije pravi 32-bitni sistem nema jednake mogućnosti rada u mreži, niti upravljanje memorijom i multitasking-funkcija kao OS/2. Kraj DOS sra je konačno na vidiku. Nadajmo se da će nam proizvođači softvera obuduće isporučivati operativne sisteme i aplikacije čiji će zahtevi biti umereni.

A.K.

Novell Netware 4.0

Beta-verzija se obećava puno

Mada je *Netware* u oblasti lokalnih mreža bez sumnje vodeći softverski proizvod ima problema pri primanama kod mreža koje se prostiru na velikom području (WAN – Wide Area Networks) i pri upravljanju vrlo velikim lokalnim mrežama. Varzija 4.0 ovog programa trebalo bi da otkioni ove nedostatke. Veliko ja zato bilo razočaranje kaos je stigla beta-verzija. Zlobnici već tvrde da bi takvom softveru mnogo bolje odgovarala oznaka alfa nego beta. Poznavaoci tržišta se druge strana amatrazu da „Novell" nije želeo suviše rano da otkrije sve svoje karte pa stoga nije u beti-verziju ugradio sve planirane funkcije.

Opseg funkcija u beti-varzili *Netware*-a je u svakom slučaju uzak i nedovoljan. Nikakvih problema ne bi trebalo da buda pri povazivanju radnih stanica pod DOS-om dok je za *Apple* računare i mašine koje rade pod OS/2 potrebno svakako izvršiti još neke dorade. Tu je i jedan se sada prilično oporn pomoćni program za upravljanje mrežama pod *Windows*-om. Još uvek se ne zna koliko dobro će verzija 4.0 rešiti problem heterogenih PC-mreža.

Dobra ideja je implementacija automatske kompresije podataka. Biće podržani i internacionalni skupovi karaktera. *Netware 4.0* nasuprot drugačijim glasinama, nije predpostavka za korišćenje udvojenih servera. Dugo očekivani dopunski proizvod – SFT Level III koji to omogućava, može se koristiti i se pretnodnim verzijama *Netware*-a i tranutno je u fazi testiranja kod velikih muiterija. On će obezbediti optimalnu tolarancijy otkaza pri radu, pošto će dva servera biti međusobno povezana optičkim kablom. Ukoliko se jedan servar pokvari drugi će automatski preuzeti njegov posao. SFT Laval III ne zahteva da dva servera budu identična. Može se na primer kao glavni server izabrati jedan 80486 računar a kao rezervni neki 80386 računar.

„Novell" zaostaja za konkurencijom (Banyan) i po globalnim uslugama. Sa sadašnjom verzijom *Netware*-a može se preko globalna liste pozvati najviše 8 servera. Konačna verzija bi trebalo da omogući komunikaciju se 60 servera. Posebno je za upravnike sistema u velikim kompanijama bitno da imaju pristup do svih raspoloživih servera. U svakom slučaju korisnici će biti pošteđeni napornih prijavljivanja na svaki pojedinačni server. Mreža nudi strukturu sličnu stablu direktorijuma na hard-disku. Kod *Netware* verzije 2 i 3 pristup mreži je bio moguć samo ako je upravnik sistema prijavio korisnika ne određenu LAN mrežu. Ok to nije neophodno u *Netware* verziji 4.0. Nova verzija *Netware*-a trebalo bi da podržava do 1000 korsnika. „Banyan" je ipak u prednosti uz pomoć „Banyan" ovog globalnog servisa *Streetalk* na primer upravnik sistema iz Minhena može bez problema da konfiguriše mrežu u Hamburgu a korsnici imaju između ostalog i mogućnost da se prikjuče ne nerazličitije mreže širom sveta. Ukoliko bi *Novell* uspeo da ostvari iste takve funkcije, koja će za to biti i ljubazna prema korsniku bio bi to još jedan plus, uz prednost koju ima na lokalnom polju, gde Banyan loše stoji.

Ostaje da se vidi i da li će *Novell* ponuditi tzv „Burst Packet Mode" koji bi naročito trebalo da poboljša performanse mreža se velikim saobraćajem.

A.K

SOFTVER

Baze podataka

Bolji pristup: Microsoft Access

Kão gečurke posle kiše niču baze podataka *za Windows*. Do sada su ovim tržištem vladali *Omnis, Paradox* i *Superbase*. Sada se pojavio novi takmičar – „Microsoftov" *Access*.

Zajedničko svim *Windows* bazama podataka je to da kroz interaktivno ili objektno programiranje olakšavaju pristup velikim količinama podataka. Očigledno je da „Microsoft" više ne želi da zaostaje na tom polju. Sa *Access*-om i najavljenim *FoxPro for Windows* ubrzo će se ne tržištu naći dve „Microsoftova" proizvoda koja bi trebalo da prkose sadašnjoj i budućoj (*dBase*) konkurenciji.

Za razliku od *FoxPro*-a za *Windows*, *Access* nema mnogo zajedničkih tačaka sa *dBase*-om. On nudi isto tako fleksibilne strukture podataka i mogućnosti upita (*query*) kao i ostali programi koji se danas mogu naći na tržištu. Međutim u mnogim aspektima ovaj program ide novim putem. Uz pomoć *Access*-a trebalo bi da korisnik buda u mogućnosti da bez mnogo programiranje sam postavi ne noga složene bazne aplikacije. Ovo se ostvaruje pomoću različitih editora na raspolaganju su moćni generatori tabela obrazaca i upita. Uz to tu su i dotirane makro-funkcije i opežan programski jezik *Access-Basic*. Korisnik, pre svega, mora definisati neophodne tabele u svojoj bazi. Na raspolaganju su mnogi tipovi podataka (samo za tip „Datum/vrama" ih ima sedam). Već i kod definisanja tabela moguće je zadavanje različitih parametara, kao što su, na primer, provera ispravnosti unetih podataka ili standardne vrednosti za neko polje.

Za pregledan unos podataka može se upotrebiti editor obrazaca. *Access* omogućava crtanje kompletnih skranskih maski. Moguće je i njihovo kamfortnije kreiranje pomoću internih „pomoćnika" se izradu obrazaca.

Svoju jaču stranu *Access* pokazuje pre svega, u odnosu na izveštaje. Moguće je generisanje labela sa samo nekoliko pritisaka na taster miše. Formati koji su unapred zadati mogu se promeniti. Podjednako elegantan je i rad se upitima (*queries*) koji se mogu izraditi najvećim delom pomoću miša. Upiti omogućavaju proizvoljna povezivanja između tabela u bazi podataka. Ni fleksibilni upiti nisu nikakav problem pomoću automatski generisanog dijaloga, korisnik u svakom trenutku može zadati uslove po kojima se pretražuje.

Da bi se u formulare integrisala „dugmad" (buttons), još uvak nije potrebno posegnuti za programiranjem. Ona se dodaju u editoru formulara a zatim im se pomoću dijaloga pridružuju svi neophodni parametri – čak i makroi koji se se svoja strane generišu iz makro-editoru. Makro je prost skup i kombinacija stavki iz padajućih menija. Ova jednostavna metoda omogućava izgradnju kompliksnih aplikacija sa konsničkiđefinisanom radnom okolinom „dugmadima" i sopstvenim menijem.

Koma ovo nije dovoljno taj može da piše sopstvene programska module uz pomoć ugrađenog bejzika koji je u ovoj verzijl izuzetno moćan on nije samo prosta suma uobičajenog dejzika i naredbi za rad sa bazama već su tu i funkcije potrebne za razne proračune.

Program je dobro opremljen i mogućnostima uvoza/izvoza (*import/export*) podataka i posaduje filtera za sve važeće formate. Onima koji ne žele da napuste svoje stare SQL podatka u pomoć priskače nova ODBC (*Open Database Connectivity*) tehnologija. U praksi ODBC znači povezivanje sa podacima iz nekog SQL servera uz pomoć ODBC drajvera. Ukoliko je takav drajver na raspolaganju, recimo se *Oracle*, moguće je direktne koristiti *Oracle*-ove podatke i njima manipulisati *Access* dopušta i laucima da se zbliže sa pojmom „baza podataka". Za profesionalce *Access* može da posluži i kao razvojni alat.

Kojim ciljnim grupama je Microsoft namenio svoje dve baze – *FoxPro* i *Access*? *FoxPro* je zamišljen za sistemske programara koji žele da razvijaju složena aplikacije. *Access* je više korisnički orijentisane baze podataka za manje i srednja rešenje pri čemu u „Microstu" priznaju da preklapanja između ove dva ciljne grupe postoje.

A.K.

Komunikacije

Microsoft i WordPerfect: Novosti u elektronskoj pošti

„Microsoft" ubrzano radi na projektu pod šifrom „Calvin & Hobbes", programu za kreiranje obrazaca za rutinarje elektronskih poruka, koji će se koristiti uz *Windows for Workgroups*. Elektroska pošta je sve popularnija u lokalnim mežama i radnim grupama, a ovakav program omogućava korisnicima da upućuju e-mail faks pa čak i glasovne poruka (*voice mail*) ne druge nodove i mreže tako što jednostavno označe odgovarajuće rubrika u elektronskom obrascu. Program će se oslanjati na „Microsoftov" *Messaging Application Program Interface* (MAPI) skup funkcijskih i servisnih poziva koji omogućavaju kreiranje aplikacija za slanje poruka kao što je „Calvin & Hobbes".

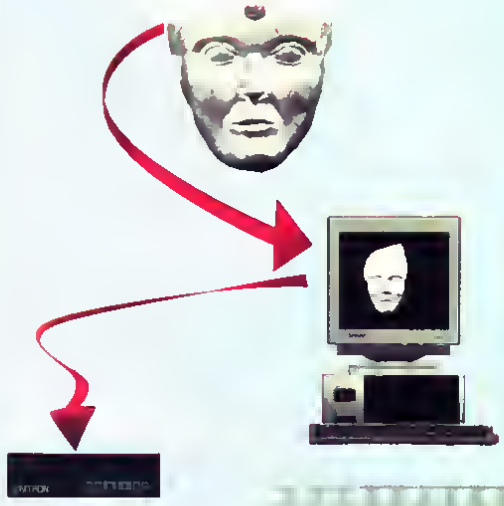
MAPI se gredi se nezavisnom arhitekturom koja omogućava pomoću samo jedne aplikacije pristup sistemima za elektronsku pošiu koji odgovaraju MAPI standardu kao i faks-mašinama i uređajima za slanje glasovnih poruka.

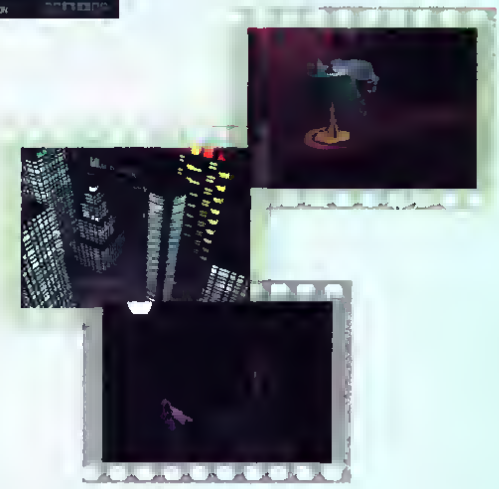
I kofporacija „WordPerfect" radi na razvoju sličnog proizvoda ali će on kombinovati elektronsku pri-

ČAROLIJA STVARANJA

Animacija

- 2D i 3D animacija
- kombinovanje žive slike i računarske grafike
- specijalni efekti
- veliki izbor materijala
- prebacivanje na video:
  - betacam
  - umatic
  - vhs





RAČUNARSKI INŽENJERING

ŠUMADIJSKI TRG 6a BEOGRAD TEL 011/ 55 78 92 FAX 011/ 55 98 45

premu obrazaca i distribuciju pošiljki. *InForms 10* će biti čvrsto integrisan sa programom *WordPerfect Office* koji ima ugrađenu opciju za elektronsku poštu. Program se očekuje početkom 1993. s sadržaće dve posebne palete dizajnerski modul će imati alatke za kreiranje i održavanje obrazaca, i direktan pristup na nekoliko poznatih baza podataka. *Filer* modul će omogućiti korisniku da popunjava formular, s radnice pod operativnim sistemima DOS *Windows, OS/2, Unix* i *PenPoint*. WP obrasci će imati i opciju „elektronski pečat", što će sprečiti neovlašćene korisnike da menjaju njihov sadržaj.

Programski jezici

Uskoro Visual Basic 2.0

Visual Basic 2.0 najavljen u novembru, obećava više značajnih novina – više kontrole više opcije ključnih reči događaja, alatki za dibagging pomagale za programiranje i – što je možda najvažnije – bolje performanse.

VB 2.0 će imati dve verzije – *Standard Edition* i *Professional Edition*. Profesionalna verzija je zamena za *Professional Toolkit 1.0* i *Control Development Kit*, dok je u standardnu verziju dodato nekoliko opcija iz paketa *Toolkit 1* – OLE kolroia za klijenta i usavršen spredšit objekat.

Alatke u obe verzije imaju dve nove opcije. Prva je grafička kontrole koje omogućavaje kreiranje raznih oblika, linija i labeli kao i ranije ali uz korišćenje menje *Windows* resursa tako da se mogu graditi veće aplikacije. Druga opcija je tzv „image box" – kao i ranij „picture box" prikazuje BMP i ICO grafičke datoteke ali omogućava i promenu veličine slike. Osim toga umesto samo 16, podržuje grafiku s 256 boja.

Sve kontrolne opcije sada su smeštene u prozor koji se može pomeriti i skrolovati (za razliku od ranijeg statičnog). Globalne i lipske definicije, ranije dozvoljene isključivo u GLOBAL.BAS datoteci sada se mogu ubaciti u svaki kodni modul (ali ne i u modul obrasca). Ako se deklaracija počne sa instrukcijom **Option Explicit** VB 2.0 zahteva da sve promenljive budu deklarisane pre upotrebe. To znači da programer mora malo više da se pozabavi strukturiranjem ali zato izvorni kod postaje lakši za razumevanje i održavanje, neročito u projektima za radne grupe.

Nekoliko novina omogućava gradnju većih aplikacija. *Visual Basic 2.0* odzvoljava četiri puta veći broj procedura po jednoj aplikaciji dvostruko veći broj promenljivih u jednom modulu, dvostruko veći broj instrukcija u jednom obrascu, dvostruko veći broj globalnih promenljivih, neograničen prostor za stringove (preme raspoloživoj memoriji i probijanje barijere od 64 KB – ali same u unapređenom *Windows* režimu). Sve u svemu *Visual Basic 2.0* za sada samo u betaverziji donosi značajne novine i vredi ga pogledati.

Sinteza glasa

Pro Voice: da programi pregovore

ProVoice vam omogućava da svojim DOS-aplikacijama dodate sintizovani govor. Paket koristi *Speech Engine* kompanije „First Byte Software" koji nizove slova brojeva i drugih podataka pretavara u govor koneteći rečnik i gramatička pravila engleskog jezika. U rečnik možete ubacili nove reči koje *Pro Voice* ne prepoznaje ili ih pogrešno izgovara.

Program Ime ugrađene alatke pomoću kojih se *ProVoice Speech Engine* vezuje se izvornim kodom programskih jezika *Microsoft C Turbo C++, Turbo Pascal Turbo Assembler* i *Microsoft Assembler*. Ove veze omogućavaju da aplikacije direktno pozivaju TSR govornu mašinu koja tekst pretvara u govor.

Cena $585

Kontakt adresa First Byte Software Torrance CA, (800) 523-2983 or (310) 793-0600 fax (310) 793-0601

Programske alatke

QuickStart 2.0: 32-bitne aplikacije u 16-bitnom Windows-u

Još jedna novost za programere – novi softver kompanije „Phar Lap Software" omogućava kreiranje 32-bitnih *Microsoft Windows* aplikacija bez napuštanja 16-bitnog *Windowsa 3.1*. *QuickStart* verzija *2.0* koristi *Win32s*, „Microsoftov" API (*application programming interface*) uz koji se aplikacije pisane za dugo očekivani *Windows NT* savršeno izvršavaju i u *Windowsu 3.1*. Upodobom *Win32s*, programeri koriste samo jedan deo kompletnog *Win32* API-ja. Po rečima kompanije „Phar Lap" to znači da programeri koji rade pod NT-om mogu koristiti sve prednosti drugih razvojnih alatki namenjenih već proverenim okruženjima. „Alatke obično zaostaju za operativnim sistemom," kaže predstavnik „Phar Lap" e Maria Vetrano. Alternative bi bile da se razvoj programa obavlja u *Windowsu 3.1* da se zatim ponovo startuje mašina i testiranje obavi u NT-u.

Postavlja se pitanje da li se može desiti da se neka aplikacija drukčije ponaša kada koristi *Win32s* pod *Windowsom 3.1* nego pod NT-om. „Phar Lap" tvr di da nisu naišli na takve probleme. Gotove aplikacije se mogu pokrenuti pod NT-om radi konačnog testiranja i otklanjanja bagova.

Za prvo vreme kompanije koje se have razvojem softvare dobiće *QuickStart 2.0* besplatno, dok se proizvod ne učvrsti na tržištu i dok se ne stekne utisak o tome „gde se uklapaju se NT-om".

Inače, Phar Lap je poznat po softveru za proširenje DOS-a koji omogućava da programi pod DOS-om rade u 32 bitnom režimu. Njihova *386/DOS Extender* tehnologije je ugrađena u najnoviju „Microsoftovu" bazu podataka *FoxPro 2.5* za DOS.

Kontakt adresa Maria Vetrano, Phar Lap Software, 617-661-1510 fax 617-876-2972)

Programski generatori

Proteus 6.0: Jednostavno kreiranje demo programa

Trideset i pet video i animacijskih efekata podrška za *Music Definition Language* glasovnu datoteku *Sound Blastera* i proširenu memoriju – to su dodaci koJime je „Genus Microprogramming" obogatio novu verziju *Proteusa* svoje prototipske/demo alatke za PC. *Proteus 6.0* nudi jednostavno kreiranje demo programa, raznih programa za obuku (tutorial) i softverskih prototipova koji kombinuju tekst i grafiku se specijalnim efektima zvukom i muzikom. Paket omogućava de se sve demo datoteke smeste u jednu EXE datoteku. te da se odredi datum posle kojeg demo više ne može da se startuje.

Cena $349

Kontakt adresa Genus Microprogramming Houston TX (800) 227-0918 or (713) 870-0737, fax (713) 870-0299.

Zanimljivosti

Microsoftov konkurs za najbolji softver

Na novembarskom COMDEX-u u Las Vegasu „Microsoft" je objavio rezultate konkursa otvorenog na svetskom nivou za najbolji *Windows NT* „sharseware" softver. Prispelo je 50 programe iz celoga sveta (među njima i jedan iz bivšeg Sovjetskog Saveza) a u žiriju su bili nezavisni novinari. Diego Aranda se „Internmeta" Steve Gibons i Brian Livingston. novinan „Infoworide" Fred Langa. urednik „Windows Magazine"-a, I Jerry Pournelle jedan od urednika Byte-a. Sponzori su bili, između ostalih "Artists Graphics" „Compuserve" „Jolt Cola" i „Microsoft". Većina aplikacije je sa Windows 3.1 III Unika prenesane na NT platformu.

Takmičenje je podeljeno u pet kategorija. e za svaku su predviđene tri nagrade. U poslovnoj kategoriji je pobedio program „While You Were Out" *Windows NT* sistem za slanje i prijem poruke kompanije „Caliente Software". Drugo mesto je zauzeo program „Moricalc" za obračun hipotečke amortizacije Roberta Paula, a treće „Time and Money Tracer" iz „Wintronixe".

U kategoriji alatki pobedio je konfigurabilni tekst editor za programere „Microemacs" Na drugom mestu je bio *Windows* komunikacioni program „Kermit" Wayne Warthene se ikonom koja predstavlje popularnog žapca Kermita iz amisije „Muppet Show", dok je treći bio simulator strujnog kola, „Nutmeg-32" Roberta Zeffa.

Prvi u kategoriji uslužnih paketa je bio „Winbatch", *batch* jezik za NT Wilsona Windowwarea. Drugi je bio „4dos" program koji unapređuje rad iz *Windows* NT komandne linije a treći „Trashman" program za brisanje „drag and-drop" datoteka koji je peipisao „Trigen Software GmbH"

U „Widget" kategoriji najbolji je bio „Icon Manager" komplet alatki za odlovanje sistematizaciju i instalaciju *Windows NT* ikona. Drugi je bio program Norberta Unterberga „Winmod" zvučni plejer datoteka koji pokreće datoteke iz Amiga MOD formata a treći „Prime32" softver koji izračunave i šalje na ekran sve primarne 32-bitne brojeva.

U poslednjoj kategoriji su ocenjivane igre. Pobedila je „Klotz" (još jedna igre na temu blokova koji padaju) druga je bila popularna igra se uklapanjem pločica, „Mah Jongg" se novim odlicima ploča novim putokazima i srednjevekovnim simbolima. Treće je bila varijanta kartaškog „pasijansa" "Thieves and Kings"

Pobednici su dobili *Artist Graphics Winsprint 160* video-karticu i *Nanao 340i* 15-inčni monitor. Druga nagrada je bio *NEC Multimedia Upgrade Kit*, a treća *Artist Graphics Winsprint 100* video-kartica.

„Microsoft" je zakazao novo takmičenje čiji će pobednici biti odjavljeni na proleonom COMDEX-u a zainteresovani kandidati detalje tonkursa mogu naći na Intranetu III Compuserve Microsoft Developers Forumu.

MULTIMEDIJA

Prepoznavanje govora

IBM-ove mašine umeju da slušaju

Tokom jesenjeg COMDEX-a IBM je predstavio četiri proizvoda koji odgovaraju na davnašnju želju korisnika – da kompjuteri razumeju ljudski govor. Dva su rađena po licenci pionira u oblasti prepoznavanje govora, kompanije „Dragon Systeme". dok je dva razvio sam IBM.

IBM je otkupio licencu za *DragonDictate-7K*, sistem za diktiranje za DOS računare. Pomoću ovog softvera, koriscici kontrolišu DOS i mnoge aplikacije govorom e na kucanjem naredbi. Ima vokabular od 7 000 reči i rečnik od 100 000 reči, e reaguje na svaki glas.

Druge licenca je za *Dragon Talk To Plus* paket za prepoznavanje govora za *Microsoft Windows. Talk-To Plus* Ime biblioteku ugrađenih komandi, e istovremeno mogu biti aktivne 64.

Oba „Dragon" proizvode zahtevaju PC sa 386SX ili močnijim procesorom i zvučnu karticu – pomiriu se IBM-ov *M-Audio Capture* i *Playback Adapter* (M-ACPA) kao i „Creative Lab"-ov *Sound Blaster 16. Talk To Plus* će počevši od marta prodavati i IBM i Dragon, po ceni od $149. e *DragonDictate-7K* po ceni od $2 295.

IBM je već ranije uzeo licencu za *Draganav" VoiceType* softver koji služi kao zamena za tastaturu.

IBM je najavio i seriju *Speech Server* softvera za RISC System/6000 radna stanica. *Speech Server* radi i na RS/6000 serveru, dok PC mašine rade pod OS/2 kao kliienti. Po rečima IBM-ovog predstavnika *Speech Server* je izuzetno precizan sistem koji prepoznaje samo glas određenih govornika i ima vokabular od 20 000 reči. Može pratiti diktat do 70 reči u minuti, e rečnik se može prilagoditi potrebama korisnika.

Speech Server će imati cenu od $6.950 za server program. $695 za OS/2 kliientski softver i $1 900 za RS/6000 karticu za ubrzanje govora. Zaseben softver za RS/6000 koji radi se IBM-ovom AIX varijantom Unixa će koštati $2 495 i očekuje se u maju.

Konačno IBM je tamsirao i *Continuous Speech Series Developer's Program*, koji nezavisnim proizvođačima softvera nudi alatke za ugradnju opcije za prepoznavanje govora u pakete pisane za OS/2 i RS/6000. Softver ima aktivni vokabular od 1 000 reči, iz postojećeg baze od 20 000. Ove alatke će se prodavati po ceni od $3 995.

Kontakt adresa Tara Sexton IBM, 914-766-3781 or 914-642-4662 Esther Agenis, Dragon Systems, 617-995-5200)

IMTEL
IMTEL
IMTEL

Alatke

Norton Commander: štek alatke za OS/2 2.0

„Symantec" je predstavio seriju Norton uslužnih alatki za IBM-ov grafički operativni sistem OS/2 2.0 *Norton Commander* je slična programu *Norton Desktop for Windows* i omogućava upravljanje datotekama pronalaženje datoteka i podršku za mrežne menije

Program nudi i upravljanje sistemom datoteka visokih performansi (HPFS)

Alatka za pronalaženje datoteka se globalnim (džoker) znacima (global file find utility) pretražuje drajvove ili datoteke po imenu ili ekstenziji putem džoker-znakova iz DOS-a, kroz cell drajv hard-diska. Datoteke koje odgovaraju kriterijumu pretraživanja se prikazuju u listi odakle se mogu startovati ili pregledati

Korisnici mogu pregledati editovati, kopirati, preimenovati premestiti ili izbrisati datoteku ili grupu datoteka uz pritisak samo nekoliko tastera ili klikova mišem Moguće je i poređenje fajlova i premeštanje datoteka iz jednog foldera u drugi Mogu se koristiti *read-only* meniji unutar računarske mreže i slati na druge OS/2 kompatibilne mreže

Da bi se padaci bolje organizovali, jedan meni se može smestiti unutar drugog DOS i *Windows* aplikacije koje rade pod operativnim sistemom OS/2 se mogu unapred konfigurisati tako da se startuju sa minimalnim makeimalnim ili normalnim prozorima.

Maloprodajna cena se *Norton Commander* je $149

Alatke

PKZip za Windows: Arhiviranje iz Windows-a

Korisnike *Windows*-a koji koriste *PKZip* za kompresiju i dekompresiju datoteka kompanija „Spiffy Software" iz Sietla će obradovati uslužnim programom *Drag 'N' Drop for PKZip*, koji će im omogućiti da arhiviranje obavljaju iz *Windows*a bez prelaska u DOS

Programu *Drag 'N' Drop* se pristupa iz *Windows File Managera*, klikom na datoteku ili grupu datoteka, koja se zatim pravuče na ikonu koju animira korisnik i koja ima nasilnan jestem-zatvarač (zipper). Datoteka će biti automatski komprimovana ili dekomprimovana. Kod kompresije jestem zatvarač na ikoni je zatvoren, a kod dekompresije otvoren. *Drag 'N' Drop* se izvršava u pozadini a korisnik dobija obaveštenje ka da je operacija završena. Ako komprimovana datoteka sadrži nekoliko datoteka, korisniku se nudi opcija da snimere samo određene delove za dekompresiju i da odredi gde da se smeste

Kada je datoteka „zipovana" (komprimovana ili dekomprimovana) smešta se u privremeni zip-fajl Korisnik ovaj korak može preskočiti, navodeći ime određene datoteke i direktorija.

„Spiffy Software" preporučuje da se *Drag 'N' Drop* smesti u startup a obzirom da zauzima samo malo memorije da bi se automatski učitavao.

Da biste koristili *Drag 'N' Drop* treba vam neka verzija PKZipa, koji se može „skinuti" se gotovo sve bolj BBS-a. Cena za *Drag 'N' Drop* je $24.95

TEHNOLOGIJA

Mikroprocesori

Priču se o Intelovoj „šestici"

Još se nije pojavio famozni "Intelov" mikroprocesor P5, a već su procurile vesti o njegovom nasleđniku – P6. Kažu da će se „šestica" pojaviti za 18 meseci i da će imati 10 miliona tranzistora – triput više nego petica i ceam puta više nego 486DX Ako se predviđanje ostvari P5 će biti najkratkotrajnija generacija procesora u istorji Intela 286 je dočao 44 meseca posle 8086, 386 je usledio za još 44 meseca, 486 za 42 meseca, dok se P5 očekuje oko 45 meseci posle 486-sice

Memorija

Fleš-memorija: jedna dobra i jedna loša vest iz Intela

Intel je još pre nekoliko meseci najavio da će 8-megabitni fleš-memorijski čipovi kasniti oko godinu dana. Sada se rok opet pomera, zbog velike potražnje, uglavnom od strane firmi koje proizvode PCMCIA (PC Memory Card International Association) kartice se skladištenje za najnoviju generaciju minijaturnih, ručnih kompjutera.

Dok tržište obiluje 1-Mb fleš-memorijskim čipovima „Intel" je jedini proizvođač koji nudi čipove većeg kapaciteta Fleš-memorija je veoma privlačna, jer daje povećan kapacitet na manjem prostoru, troši malo struje i čuva sadržaj čak i kada se struja isključi

Neki analitičari su predvidali da će fleš-memorija zameniti hard-disk drajvove. Međutim, da bi se to desilo fleš-memorija bi morala biti manja, se više kapaciteta, i jeftinija

„Intelov" predstavnik Majki Salven je nedavno neviniarina saopšto dve vesti – jednu dobru i jednu lošu Dobra vest je ogroman rast potražnje – za poslednjih pet meseci povećana je za 500% Loša vest je to što proizvodnja ne može da prati toliku potražnju Muku muči Intelov japanski snabdevač „NMB Semiconductor" tako „Intel" stalno šalje nove ljude u Japan Saliven je rekao da se planirani datum isporuke kraj 1992. odlaže za godinu dana. Zašto toliko zakašnjenje? Za sada 8 Mb čipova proizvodi samo „NMB Semiconductor" ali se razmišlja o početku proizvodnje i u Intelovom postrojenju „Fab 7" u Santa Klari dok se Intelvremeno vrli veliki pritisak na „Sharp" a kojim „Intel" već ima ugovor o proizvodnji 8 Mb čipova

Kašnjenje jeste neprijatno ali u „Intelu" veruju da neće imati finansijske reperkusija a obzirom da je udeo svih memorija, pa i fleš-memorije, zanemarlv u odnosu na profite od mikroprocesora (u Intelovom slučaju 10%) Ali predviđanje govore da će tržište fleš-memorija naglo procvetati i se sadašnjih 130 miliona dolara do 1995 porasti na 1,5 milijardu

„Advanced Micro Devices" i „Fujitsu Limited" najavljuju svoje 8-inčna fleš-memorijske pločice (wafers), koje bi se mogle pojaviti 1994. a omogućile bi proizvodnju čipova od pola mikrona, pa i manje.

I IBM je najavio plahove za utrostručenje memorije na istom prostoru takozvano „ubijranje" pomoću tehnologije razvijene u Kalifornijskoj firmi „Irvine Sensors"

PERIFERIJE

Tastature

Telefoniranje preko tastature

CampuPhone, PC-kompatibilna tastatura se 101 tasterom firme „Integrated Technology" ima jednožičnu telefonsku liniju i interfejs za slušalice se mikrofonom, tako da nije potreban poseban telefon za telekomunikacije Konsti se numerička tastatura umesto telefonskog brojčanika, a postoji ugraden softver za automatsko okretanje broja.

Cena $299

Kontakt adresa Integrated Technology, Inc , 76 South Orange Ave , South Orange, NJ 07079 (201) 907-0200, fax (201) 762 7234

Štampači

Star: Male, jeftine, Windows kompatibilne

Firma „Star" proširila je liniju svojih proizvoda izbacivši na tržište dve nova i jeftina modela – *LC24-100 Mulitfont* i *LC-100 Colour* Ova dva modela se po svojim mogućnostima nalaze na donjem kraju „Starovog" asortimana Uprkos tome, oba nude interesantne mogućnosti 24-igličnl štampač LC24-100 podržava maksimalnu rezoluciju od 360x360 tačaka po inču i nudi čitav niz fontova Štampač LC *100* je 9-igličnl se maksimalnom rezolucijom od 240x216 tačaka po inču ali zato može da štampa u boji

LC24-*100* se odlikuje izuzetnom podrškom *Windows*-u Sa ovim modelom se isporučuje disketa na kojoj se pored četiri *TrueType* fonta i različitih drajvera za standardna aplikacije nalazi i jedan specijalno prilagođen drajver za upotrebu u *Windows* okruženju On ne samo da optimalno upravlje štampačem, već i komprimuje podatke pre prenosa do štampača, što ubrzava štampanje i smanjuje opterećenje sistema prilikom štampanja. LC24-100 može da prenese na papir 5 NLQ fontova u deset varijacija. Kao što je kod matričnih štampača uobičajeno, tu je samo jedan draft-font Brzina štampe koju navodi proizvođač se kreća oko 64 znaka u sekundi u NLQ režimu a u običnom draft režimu se penje na 192 znaka u sekundi

LC-*100* štampa nešto sporijim tempom On štampa 180 znakova draft kvaliteta i 45 znakova NLQ kvaliteta u sekundi Četiri NLQ fonta u sedam varijanti omogućavaju pristojnu pismenu korespondenciju i se ovim štampačem Ono što LC-100 čini interesantnim za korisnike-amatere je mogućnost štampanja u boji i se njim se dobijaju drajveri za većinu standardnih aplikacija

Firma „Star" nastoji da upotrebu svojih štampača učini što je moguće više jednoobraznom Podešavanje oba štampača je veoma dobro rešeno Prekidač za uključenje štampača nalazi se kod oba modela levo na prednjoj strani Oba štampača imaju po četiri tastera dovoljna za sva podešavanja. Preko njih se podešava veličina fontova i upravlje se papirom Devet svetlećih dioda simulira mrske DIP-prekidače koji se sad mogu lako podešavati pomoću ona četiri tastera. Ipak, objašnjenja u priručniku nisu dovoljno jasna, tako da se od novog korisnika zahteva izvesno predznanje o štampačima.

Cena modela *LC24-100 Multifont* je 798 DEM dok model *LC-100 Colour* košta 648 DEM Mogu na ručitl od Star Micronics, Frankfurt A.K.

ZAŠTITE

Virusi

IBM i Western Digital: PC imun na viruse

IBM planira predstavljanje antivirusnih personalnih računara, zasnovanih na novoj hardverskoj tehnologiji koju je uveo „Western Digital" Primenjena na standardnu arhitekturu (ISA) se WD7855 kontrolerom u viou jednog čipa, ova hardverski zasnovana tehnologija predstavlja preventivni lek protiv virusa.

„Western Digital" kaže da se ugrađena „imunizatoteka" logika za kontrolu sistema pokreće preko SMI-a (*System Management Interrupts*) i zatim preko WD7855 kontrolera prati sva zapise na hard-disk, reagujući na svaki sumnjiv postupak.

Ovaj Imunizator (*Immunizer*) zabranjuje upis u sva regije hard-diska u kojima se nalaze izvršne datoteke jer se upravo to datoteke u koje obično prodru virusi preuzimaju kontrolu ned centralnim procesorom i počinju svoj prljavi posao *Immunizer* štiti i *boot* sektor hard-diska što je posebno važno Naime statistički podaci koje je objavio IBM ukazuju da se ogromna većina „divljih" virusa javlja upravo u boot sektoru. „Western Digital" naglašava da WD7855 kontroler može sprečiti virusnu infekciju a ne samo otkriti virus. Inače program otkriva i poznate i nepoznate viruse i ne utiče na sistemske operacije i performanse U kombinaciji sa konvencionalnim softverskim antivirusnim paketima, WD7855 kontroler obezbeđuje zaštitu od sve šire epidemije postojećih novih i multiramih virusa

Ova tehnologija će biti uključena u sve buduće sistemske kontrolera ove kompanije uključujući i 32-bitne IBM će ugraditi „Western Digital" kontroler i u svoju novu desktop liniju se IBM-ovim 486SLC2 mikroprocesorom

Pripremila: Ranka Jovanović

YU SLOVA
EPSON, STAR, PANASONIC
I SVI OSTALI *ŠTAMPAČI*,
LATINICA, ЋИРИЛИЦА
011/685-779

021 369 907

APLIKATIVNI SOFTVER SA HARDVEROM

BANKARSTVO KNJIGOVODSTVO POLJOPRIVREDA OPSTI PROGRAMI

Enter®

kompjuterski inženjering i AOP

Novi Sad, Narodnog fronta 53

Pokrenite svoje računare!

DEMINO

Posle velikog

uspeha verzije 1.0 nova, poboljšana verzija

PLATE 2.0 stoji Vam na usluzi.

Mogućnost obračuna na bazi tri vrste bodova, fiksnih i varijabilnih iznosa, satnice... ● Uključivanje u obračun da 100 različitih vrsta primanja i obustava ● Neograničen broj isplata u toku meseca i godine ● Praćenja primanja po dva osnova: obračunski i isplatni period ● Dokumentacija za isplatu u SDK: ZID, specifikacija virmani... ● Dokumentacija firme: platni spisak, virmani za uplatu po bankama, isplatni listići radnicima, arhivska dakumentacija... ● Dabijanje četiri različite vrste proseka ● Brz obračun za veliki broj radnika ■ DODATAK: instalacija, obuka, prvi obračun

Pogled u budućnost!

jugodata

SOFTWARE

8 Revolucije 326, 11050 Beograd, tel/fax: (011) 418-326, 419-768, (081) 11-618

LEP, SNAŽAN, KOMPAKTAN

U prošlom broju predstavili smo vam SGI Iris Indigo – jednu od UNIX radnih atanica firma koja je orijantisana na 3-D grafiku. Međutim, najveći deo UNIX grafičkih radnih stanica na svetskom tržištu prodaje druga firma, kalifornijski „Sun", čiji je vrtogiavi uspeh zasenio čak i imena kao „Apple" ili „Microsoft". Praktično, „Sun" je osmialio i materijalizovao pojam grafičke redne stanice, njenog povazivanja i umrežavanja, i zato mu pripada zaslužno mesto u istoriji računarstva.

Od mnogobrojnih modela zasnovanih na SPARC porodici RISC mikroprocesora, nazvanih SPARCstations, u ovom broju ćemo prikazati jedan od najatraktivnijih i najkompaktnijih – *SPARCstation IPX*

Nebojša Novaković

NA PRVI POGLED

SPARCstation IPX je, najpre, izuzetno lep računar. Atraktivan dizajn je i inače odlika većine UNIX grafičkih radnih stanica, ali Sun modeli se baš posebno ističu. Izvanredni dizajn SPARCstation serije je kao i NEXT stanice i NEC FG monitori delo poznate dizajnerske kuće „FrogDesign" Za razliku od *SPARCstation 2*, koji je u sada već čuvenom i naveliko kopiranom „pizza-box" kućištu, krem kućište IPX je kao što na slici vidite „lunch-box" oblika. Pri tome je usklađene identične rupičasta tekstura prednje strane kao i na *SPARCstation 2*. Interesantna činjenica je da se „Sun"-ovi spoljni dodaci (diskovi, trake CD ROM itd.) po visini uklapaju sa *SPARCstation 2*, a po širini sa IPX-om tako da je lepo uklapanje moguće po vertikali ili po horizontali

SPARCstation IPX u svom malenom kućištu (9 6"x10 4"x4 6") sadrži osnovnu ploču sa 40 MHz SPARC procesorom i koprocesorom 64 KB keša 16 do 64 MB RAM 1 44 MB flopi i 207 ili 424 MB herd disk Na osnovnoj ploči se, pored procesorskog dela i memorije, nalazi i ultrabrzi GX grafički akcelerator koji postiže 480 000 2-D i 310 000 3-D vektore u sekundi – dovoljno za *real-time* 3-D manipulaciju žičanih modela, što i jeste jedna od namena IPX-a Te performanse će za PC svet ostati san čak i sa najbržim *local bus* grafičkim karticama sadašnje generacije Sam procesor inače, postiže pristojnih 24 SPECmarks (SPECmark je, ukratko rečeno, VAX-MIPS koma se može verovati, sabran iz više realnih aplikacija) Te performanse u ne-grafičkim aplikacijama predstavljaju brzinu oko 30% veću od 486DX2-66 klase u celobrojnom i oko 2 puta u FP delu (grafičke aplikacije postižu znatno veća ubrzanje čak i bez ikakvih dodatnih akceleratora, jednostavno zbog daleko naprednije RISC arhitektura) Se svojih 136 registara opšte namene simetričnim setom instrukcija i snažnom podrškom multitaskingu SPARC generacijski tuče Intelovu 80x86 porodicu

Memorija je u posebnim 33-bitnim (paritetni bit po reči) SIMM modulima, od kojih svaki ima kapacitet 4 ili 16 MB, zavisno od toga da li koristi 1 Mb ili 4 Mb memorijska čipova Sun SIMM standardi u najvećem broju slučajeva ne odgovaraju PC ili PS/2 SIMM standardima, tako da ne možete vršiti razmenu memonje između njih. To inače važi i za ostale proizvođače grafičkih radnih stanice

Na polju interfejsa, IPX je solidno opremljen ugrađeni su SCSI-2, *Ethernet*, 2 X 25 kompatibilne RS-232/RS-423 porta sa posebnim konektorima koji se spoje izvode u standardne preko adaptora, audio I/O portovi kao i portovi za FDD, tastaturu i miša. Na ploči su i dva *Sbus* ekspanziona slota brzine 80 megabajta u sekundi *Sbus* pločice su znatno manjeg formata nego AT na primer ali zato postižu znatno bliskije povezivanje sa osnovnom pločom nego što je to slučaj kod PC-ja, izuzev u *local bus* verzijama Na raspolaganju je nekoliko stotina *Sbus* kartica najrazličitijih namena

Ugrađeni zvučni potencijali nisu ništa posebno u odnosu na *IRIS Indigo*, all su u svakom slučaju izvrsni u poređenju s onim što još uvek standardno nudi PC Audio ulaz i izlaz „telefonskog" kvaliteta je podržan sa 8-bitnim uzorkom i 8 kHz uzorkovanjem Perfenje podržane na audio-portu su dinamički mikrofon kesetofon slušalice i spoljno pojačalo

SA SOFTVERSKE STRANE

SPARCstation koristi *SunOS*, veoma raširenu „Sun"-ovu verzsju UNIX-a koja je odnedavno doživela novu mladost pod imenom *Solaris* Novi *Solaris 2*, koji objedinjuje puni multiprocesorski UNIX V 4 sa dodacima i izvrsno *OpenWindows* grafičko okruženje raspoloživ je sada i za Intelove 32-bitne 80x86 procesore

Ukratko, rečeno Solaris sa svojih 130 MB koje korisnik dobija na CD-ROM je najviše „Mac-like" UNIX i, u isto vreme, treći operativni sistem (prvi u UNIX svetu) po rasprostranjenosti na svetu posle DOS-a i MacOS-a

OpenWindows je „Sun"-ova mrežne prozorska okolina koja podržava puni multitasking i distribuirano procesiranje više poslove za više korisnika preko više računare u mreži – aplikacija mogu raditi u realnom vremenu na nekom udaljenom računaru u mreži a biti prikazivane i upravljane lokalno sa mišem Uz *OpenWindows* dolazi *DeskSet*, set priručnih aplikacija nalik onome u *Windows 3 1*, koji sadrži *File Manageram TEXTEDIT Debug, Calendar Manager*, sat *Calculator, Print* alatku, *Mail, Snapshot - hard-copy* ekrane, *Icon Editor Performance* merači me, kao i kompletan *on-lme idiot proof* udžbenik Za razliku od DOS-a, *Solaris* ime ugreden ceo mrežni operativni sistem, napredniji i snažnija i od *Novell Netware 3 1* – bes pistno, bez obzira na broj korisnika! Uostalom, *Sun NFS* (*Network File System*) je svetski standard za mreže

Za razliku od „Silicon Graphics"-a, „Sun" radne stanice nisu isključivo usmerene na grafiku – one su jednako orijentisane i na tržište brzih i kompaktnih *database* i *file* servera sa višekorisničkim i mrežnim bazama podataka (ORACLE Informix, Ingres) kao i snažnih višekorisničkih računara („supermikro") sa i više stotina terminala Slične gradacije takođe postoji i kod IBM (*Bull* takođe) HP i DEC

„Sun"-ove SPARC mašine su u svetu trenutno Nr 1 na polju razvoja softvera Kao karakterističan primer navešćemo da se *AutoCAD* i ORACLE razvijaju na *Sun SPARC* stanicama a posle prebacuju na sve ostale platforme Praktično sav PC-softver po PC cenama a znatno boljim performansama radi na SPARC platformama a tu su još i stotine ultraozbiljnih naslova sa naučno-inženjerskografičkih područja radnih stanica – gotovo 5,000 proizvoda

POGLED UNAPRED

Zahvaljujući novom *TI SuperSPARC* procesoru, osetno bržem i od još nerođenog 586 nova generacija ultrabrzih multiprocesorskih SPARC radnih stanica i servera već je ugledala svetlo dana Postižući preko 300 MIPS iz samo 4 paralelna procesora. SPARCstation 10 sablja celih 512 MB RAM i preko 4 GB disk kapaciteta u „pizza-box" kućištu A tokom dolazeće godine pojaviće se i prve prave 64-bitne SPARC stanice

Korisna adresa:
AEROINŽINJERING
Bulevar Nikole Tesle 42A
Tel 698 600, fax 691 862

Preko 300 MIPSa Četiri paralelna procesora, 512 MB RAM-a i 4 GB u tvrdim diskovima

MI VAM NUDIMO IZLAZ

Oblast računara sve više liči na **lavirint**. Koji hardware, koji software, namenski program, prateca literatura, nameštaj, kako sve to povezati u sistem koji funkcioniše?

CET COMPUTER SHOP nudi Vam hardware različitih proizvodača. Mi smo ovlašceni dileri svetskih proizvodaca software-a (**Microsoft, Borland, Autodesk, Lotus**...). Uz našu pomoc možete kupiti bilo koji licencni software. CET COMPUTER SHOP nudi veliki izbor domace i strane literature iz ove oblasti, kao i potrošnog materijala i rezervnih delova. Kod nas možete da izaberete i kupite specifičan nameštaj. U CET COMPUTER SHOP-u imamo još nešto. Mi posedujemo **TIM KVALIFIKOVANIH LJUDI** koji ce Vam pomoci da formirate sistem koji funkcioniše.

Computer Equipment & Trade

11000 Beograd • Skadarska 45 • Tel. 343-043

Design: Labyrint • Priprema: B&Z

SILA SLICI DAJE UBRZANJE

Generisenje i prikaz slike epada u najteže poslove u grefičkim eplikecijema (od računere se zahteva i manipuiisenje ogromnim količlname podataka i intenzivno rečunanje). Zato ne čudi što proizvođeči PC hardvera ave taža prate zahteve u kompjuterskoj grafici i multimediji (*true color, photo realistic, real time visuelisetion* – prirodan izgled predmete po boji i obliku, uz trenutnu pojevu slike na ekranu). Grafička kertica *Superstation 3D GB860*, krunski model u čitavoj familiji grafičkih ekceleratora firme Hercules, ima ambiciju de ne PC pletforme donese prirodnu sliku, brzinom – superkompjutere.

Jovan Regusek

Svojim izgledom *Superstation 3D GB860* izaziva u najmanju ruku, strahopoštovanje. Dve ogromne procesora u paraleli (Intelov 64-bitni i860 RISC procesor na 33 MHz i Texas Instruments 32-bitni grafički procesor TMS34020 na 32 MHz), 18 MB programske i 2 MB brze video memorije i oblije praćeih čipova koji pokrivaju celu površinu šesnaestobitne ISA kertice punog PC formata nagoveštavaju da se radi o grafičkoj mašini za najteže CAD poslove. Izdi i upanvanje procesora (i860 ima paiplajn ehniekturu ze 3D grefiku i 3D instrukcije u hardveru i premešta 50 megapiksela u sekundi dok TMS34020 ima 2D instrukcije i raster u hardveru e piksele premešta brzinom od 20 megapiksela u sekundi) svedoče o želji firme Hercules de ponudi totalno rešenje na polju grafičkih akceleratora. Kertica obezbeduje oblje grafičkih formale do 1280×1024 tačica i do 16 7 miliona odje uz osveža-vanje slike brzinom većom od 72 puta u sekundi. Nedevno smo imali zadovoljstvo da sa ovom kertlcom svakodnevno radimo gotovo dva meseca i tako proverimo kake RISC procesorske tehnologije funkcionišе u praksi – i to na najtežim poslovima.

Superstation 3D je projektovana da radi kao grafički akcelerator u mašinama 386 i više klase. I ne može se pokrenuti čak ni pod DOS-om dok se ne izvrši softverska priprema kertice i mašine u koju se ugrađuje i ne installiaju TIGA drajveri. To znači da se ugiadnjom ove kertice ne možete odreći usluge standardnog grefičkog adaptera. VGA klase. Za VGA adapter je na ovoj „superstanici" kao i kod svih specijalizovanih grafičkih kartica predvidene VGA prevodnica (*pass through*) koje sistemu obezbeduje potpunu softversku kompatibilnost, pri čemu ove kertice bez poteškoća dele jedan monitor. *Superstation 3D* se povezuje se monitorom preko BNC priključaka sa posebnim vodovima za RGB boje i signale za sinhronizaciju slike. e VGA (u našem slučaju na bazi čipa ET4000) ne uobičajeni način, preko standardnog D-

WINDOWS: ŽIVE SLIKE

Kod prikaza u DOS-režimu između ovih kartica, naravno neme nikakvih razlika. All prelazak u *Windows* je. doslovce preporodio jedan (ni po čemu izuzatan) kolor monitor od 20 inča – bez obzire na odabranu rezoluciju boje su bile življe a likovi slove mnogo oštriji. *Windows 3 1* je jedna od dve ciljne aplikacije za ovu karticu i zato je softverski veome bogato podržne – na listi drajvera nalazi se pelnaesti grafičkih formeta za rezolucije od 1280×1024 tačkica u 256 od 1024×768 tačkica u 32 768 boja i od 768×576 tačkica u 16 7 miliona boje. Prelazak sa režima na režim u verziji koje je name bile ne raspolaganju obavlje se na mukotrpan *Windows* način preko **setup/eptions/display** menija se obavaznim restartom programa. Nedavno smo saznali da je u pripremi nova verzija drajvera koja omogućuje izbor režima u letu. na pritisak tastera, bez napuštanja aplikacije.

U softverskom paketu nalazi se i program *Image-Prep* za obardu digitalnih fotografija u boji. Ove sletke (koju topto preporučujemo grafičarima i dizajnerima) na ubedljiv način pokazuje kolorističke mogućnosti pa i brzinu *Superstation 3D* kartice. Imali smo utisak da bi se kapima rose na rascvetaloj crvenoj ruži mogla ugasiti žed – toliko su bile stvarne. Za brzinu kažemo „pa i", jer je brzina relativna i krajnje varijeve stvar. Ne male dobitke u brzini korsnik se brzo navikava, a „zvučni zid" se izgleda ne probija baš tako

TESTOVI: PRAKSA JE NEŠTO DRUGO

Mašinski i aplikativni testovi *Superstation 3D* kartlice daju spektakularne sasvim očekivane rezultate – procesorsku snagu koja je ugrađene u ovu „superstanicu" mogu de opravdaju samo drastični dobici u brzini. Prema testovima *Windows Benchmark 2.0* (PC Magazin) kartica je u ugnsu blokove podetaka (Bitblt) brža 8 u crtanju vertikalne linje 7 e u popunjavanju pravougaonika bojem 2 puta od VGA adaptera u režimu „minimalna rezolucija / maksimalan broj boja" odnosno 20 8 i 7 puta u režimu „maksimalne rezolucija / minimalan broj boja". Ovi rezultati koji se odnose na najelementarnije grafičke operacije vredni su svakog poštovanja all ih ne teba uzeti zdravo za gotovo.

Slični rezultati se dobijaju i u programu *WinTach* koji je razvio Texas Instruments za testeranje svojih grafičkih processora u stvarnim uslovima. Program

Slika 2 Windows brzinomer

	ET4000 Superstation	Faktor
Obrada teksta	4 51	10 03	2,04 puta
CAD / crtanje	7 38	63 38	8,58 puta
Radne tabele	4 96	14 02	2 82 puta
Slikanje	6,55	12,90	1 96 puta
Prosečna brzina	6 06	25 08	4 13 puta

Aplikativni testovi pokazuju veliki faktor ubrzanja, all se ono ne oseća u stvarnej primeni, jer *Windows* aplikacije ne mogu da koriste hardverske pogodnosti kartice Superstation 3D

obuhvata četiri vrste testa – *obradu teksta* (sa intenzivnim formatiranjem i promenom pisamaj *radne tabele* (sa intenzivnim skrolovanjem modifikovanjem i grefičkim prikazom podataka) *CAD/Draw* (sa intenzivnim crtanjem zumiranjem i pomeranjem crteža) i *slikanje* (od osnovnih grafičkih elemenata do kompletne slike) – a brzinomer je na našoj test mašini (486/33 MHz se procesorskim kešom 256 K i 8 MB RAM-a) pokazivao u zavisnosti od izabrane rezolucije uvek preko 25 jedinica. Pod potpuno istim uslovima, ET4000 kartica u susednom slotu je razvijala prosečnu brzinu od samo 6 jedinica. U svim aplikacijama *Superstation 3D* je bio 2 do 3 puta brži osim na CAD testu gde je umakao u brzini za čitavih osam puta. Ne znamo kolike je ovaj brzinomer pouzdan all su i ovi rezultati s obzirom na namenu kartice i na ugrađene resurse sasvim očekivani i – logični.

Neme medutim bojazni da će se korisniku kad startuje neki *Windows* program, od ove brzine zavrteti u glavi jer *Windows* ne ume de papučicu gasa pritisne do daske. Korisnici programa *CorelDraw* verovatno najpopularnijeg paketa za grafički dizajn u *Windows* okruženju biće razočarani kada vide da *Superstation 3D* poznatu demo-sliku „Kolibe pod snegom" crta umesto očekivanih osam puta brže nešto (5 sekundi sporije od najstandardnije VGA kartice (97 86 sekundi). Ne druge populame aplikacije kao što su *Ventura* ili *Excel* koje vapiju za brzim prikazom nece procvetati se ovom kertlcom. U čemu je tajna? U tome što *Windows* aplikacije nisu prilagođene za ovu grafičku mašinu. Umesto da kartici proseli listu poslova i njoj prepusti sve radnje oko generisanja i osvežavanja slike *CorelDraw* sam crta sliku, ostanjajući se na procesor u računaru da bi je posleo samo ne pnkaz kao što to čini i sa bilo kojim drugim video adapterom. On, dakle vozi „porše" na isti način kao i

Nastavlja se na strani 97

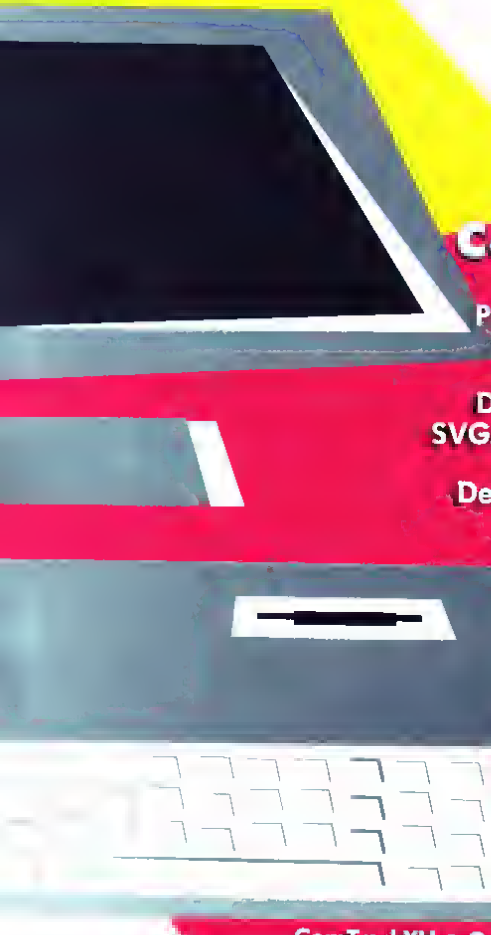
Slika 1

	Standardna VGA 640 x 480 16 boja	Superstation 3D 640 x 480 16,7 mil boja	Superstation 3D 1280 x 1024 256 boja	Faktor ubrzanja 640 x 480 16,7 mil boja	Faktor ubrzanja 1280 x 1024 256 boja
BitBlt	348 364	3 009 571	6 704 182	8 64 puta	19 24 puta
Vertikalna linija	303 352	2 110 253	2 604 954	6 96 puta	8 59 puta
Popunjavanje pravougaonika	3 328 858	6 365 623	2.174 7698	1 91 puta	8 53 puta

Brzina (piksela u sekundi) elementarnih grafičkih operacije u režimu „prirodne boje" (16,7 miliona) i „maksimalne rezolucija" (1280 x 1024), poslednje dve kolone daju faktor ubrzanja

ComTrad

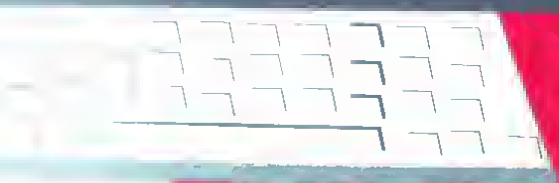
the ART of computer making

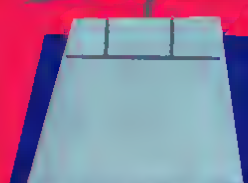


ComTrad 386SX/25

Procesor 386SX/25MHz, 2MB RAM
Floppy uređaji 1.2MB i 1.44MB
Tvrdi disk 105MB IDE
Dvo serijsko i jedon poralelni port
SVGA grofički odapter 1024x768/512kB
SVGA kolor monitor
Desktop kućište, ASCII tostoturo, miš

2,095.-





ComTrad YU • Genex apartmani, V. Popovića 6, Beograd
Telefon: (011) 222-41-51, 222-26-51 • Fax: (011) 222-41-39
Radno vreme: 9-17, panedeljak-petak

ComTrad COMPUTERS

SPRINT
Novi Sad
[illegible]avasadskog
[illegible]ma 35
[illegible] (021)
6[illegible]17
Fa[illegible]1
623-[illegible]

ComTrad Shop
Beograd
Cvijiceva 104
Tel (011)
752-663
Fax (011)
752-663

INFOTRADE
Pristina
Hekurudha 7
Tel (038) 25-830
Fax (038) 25-822

COMTRAD
KRAGUJE[illegible]
Kragu[illegible]
Kragu[illegible]ckog
aktab[illegible]02
Tel [illegible]) 60-336
Fax [illegible]4) 67-117

MANIR
Ruma
Grabljanska 2
Tel (022) 421-265
Fax (022) 421-265

OBRAČUN KOD O.K. LOKALA

Ovaj članak je odgovor na tekst D.V. Vaseljnovića „Lokal je stigao" u „Računarima" 88. Članak je primer da izreka „Lies, damn lies, benchmarks" ima puno opravdanja. Da bi se uradio test bilo koje komponente računara, potrebno je veliko teoretsko predznanje o toma šta se tastira, a naročito kako se testira i šta znače dobijani rezultati.

Dragan Đumeljić

U prikazu osnovne ploče *Headland HT340* se ugrađenim *local-bus* VGA kontrolerom HT216 dominira razočerenje autora jer je očekivao značajno bolje rezukate u odnosu na druge komponente na testu (a posebno WD 90C30 kartica). Uzroci njegovog razočerenja su u nepoznavanju mogućnosti *local-bus* tehnike u loše uradenom testiranju i pogrešno protumačenim rezultatima testiranja

Pre nego sto pređemo na analizu razultata raz motrimo kakve su teoretske mogućnosti *local-bus* tehnike i gde su ograničenja

LOKALNI-(auto)BUS ILI FERARI

Na IBM AT osnovnoj ploči postoji nekoliko magistrala npr magistrale za lokalne periferije magistrala za lokalnu memoriju sistemska ISA magistrala a ove se dalje mogu deliti na adresne kontrolne i magistrale podataka „Local-bus" označava magistralu koja se dobija priključivanjem direktno na spojne linije procesora bez kontrolera i arbitera magistrale tako da vezna logika minimalno usporava saobraćaj na magistrali Pristup lokalnim resursima je lakši i brži I radi se onom brzinom kojom to omogućavaju procesor i resurs

Koje je teoretska razlika između ISA i *local-bus* magistrala? Ako se radi samo po standardu (što je preporučena metoda) vreme ciklusa na ISA magistrali je minimalno 250 ns (uz upotrebu 0WS signala) a magistrala je širine 16 bita. Za lokalnu magistralu teoretsko minimalno trajanje ciklusa je 30 ns za 80486DX-33 procesor a širina magistrale je 32 bita Lako se izračunava da je teoretski maksimalni odnos brzine ove dve magistrale 1 16 5

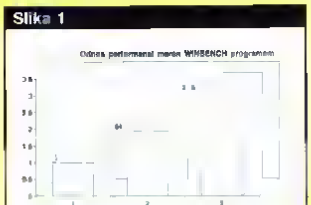
Poznato je da je minimalno vreme pristupa (po deklaraciji proizvođača) za memorije koje se koriste na VGA karticama obično 70 – 80 ns Kada se ne bi koristile specijalne tehnike pristupa minimalno vreme memorijskog ciklusa koje bi se mogio postići bilo bi oko 150 ns Kad se uzme u obzir i kašnjenje u kolima adresnih dekodera bafera i slično taj broj se poveća va na minimalno 170 ns Ako sada taj broj uporedimo se 250 ns što je trajanje ciklusa ne ISA magistrali vidimo da se neka spektakularna ubrzanja ne mogu postići

D V Veselinović je redio se taktom od 13 3 MHz ne ISA magistrali (trajanja ciklusa 150 ns) U tom slučaju je odnos takta magistrala „33 / 13 3 = 2 48" Na osnovu ovog broja D V Veselinović očekuje ubrzanje oko 1 9 puta Oveko složenom „analizom" metodom deljenja 3 kruška i 2 jabuka vrlo retko se mogu dobiti lole tačniji rezultati O zaključcima na osnovu takvih rezultata ne treba ni govoriti Moguće ubrzanje treba računati posebno za ubičajene grafičke operacija a pri tome uzeli u obzi I kako procesor 80486 može realizovati ove funkcija kao i vreme potrebno za osvežavanja ekrana

Posebna priča su mogućnosti procesora 80486 lako je trajanje takta samo 30 ns treba pokazati da procesor može izgaanasati naki koristan podatak koji bi se tom brzinom i prenosio Uobičajeno je da se najhrđim instrukcijama smatraju STOSW i MOVSW Kratak pogled u tabelu vremena izvršavanja pokazuje da se ove instrukcije izvršavaju za minimalno 5 odnosno 7 taktova. Oda sanak pusti o letečoj grafici na lokalnoj magistrali! Iako može da se govori i o problemima optimizacija koda za 80486 procesor očigledno je da spektakulernog ubrzanja neća biti

Veći dobitak se može postići time što je lokalna magistrale 32-bitna ali su za sada vrlo retki VGA kontroleri koji ovo podržavaju Pojavili su se u poslednjih

Slika 1



Odnos performansi meren Winbench programom

nekoliko meseci i očekujemo da ih testiramo za nekoliko nedelja

Pokažimo samo još neke smernice koje bi omogućile realnije sagledavanja problema Razvijeno je dosta tehnika za ubrzavanje rada se video memorijom Nabrojaćemo nekoliko osnovnih

1 Upotreba memorijskih kola koje omogućavaju stranični pristup Kade se prvi put pristupa podatku koji je u okviru jedne stranice obično oko 2000 memorijskih lokacija pristup traje oko 170 ns Ako se ponevlje pristup lokaciji unutar iste stranice vreme ciklusa se može značajno smanjiti na približno 80 ns

2 Upotrebom „cache" sistema za prikazivanje slike, ostvaruje se rasterećenje video memorije Podaci se očitsavaju u blokovima tako što se koriši stranična tehnika očitavanje memorija između čitanja ovih blokova, memorija je potpuno slobodna za korišćenje

3 Upotreba „cache" sistema za pristupa procesora Svaki upis se vrši u interne registre VGA kontrolera a stvarni upis u memoriju kad je to moguće Ove tehnika funkcioniše samo kad se podaci na prenose u blokovima

Kakva ubrzanja u radu se VGA karticom možemo očekivati u slučaju korišćenja *local-bus* tehnologija na primeru nekih češćih grafičkih operacija?

Greške u članku „Lokal je stigao"

1 Standardna sabirnica računara ne postoji već sistemska, koja može biti ISA ili EISA (1 2)

2. AT Bus kartica ne znači ništa (2 6)

3 Linearna veza između procesora i video podsistema ne postoji možda direktna veza dok je linearna zavisnost (7 1)

4 Takt od 50 MHz nije problematičan ploče se tim taktom se teže prave relativno su skupije ali je ta tehnologija savladana Te ploče obično imaju takt od 50 MHz dok ploče sa procesorom od 33 MHz obično imaju takt od 66 MHz (2 9)

5 To što je „referentni oscilator 50 MHz" ne znači da su u obzir uzeti VESA modovi niti ime bilo kakva veza sa VESA standardom (5 12)

6 To što je Mujac čip brz na piše se se 80 ns već 80 MHz (5 16)

7 Grešaka ima još ali se mogu podvesti pod drugi deo

1 Upis bloka podataka u video-memoriju (npr za crtanje prozora) Jasno je da je ovde ograničenje brzina procesora Uz upotrebu stranične tehnika rada, može se očekivati minimalno vreme prstupa oko 80 ns el procesor na može održati korak pa je očekivano vreme ciklusa barem 120 ns

2 Crtanje linije Ovo je mnogo složenija operacija od direktnog upisa u video memoriju Pekrazumeva izvršavanje algoritma za crtanje (obično Bresenham) koji kad se ude u petlju crtanja ima u proseku oko 5 mašinskih instrukcija dve prstupa memoriji i jedan pristup I/o registru Kad se uzme u obzir da se u praksi obično nalazi i poziv funkcija razna testiranja oblasti crtanja („clipping"; onda se pokazuje da je približno jednako vreme koje se utroši na rad procesora i vreme potrebno za pristupanja VGA kartici Upotrebom *local-bus* tehnika može se smanjiti samo ovo drugo vreme Odmah se može videti da nisu moguća neka spektakulerna ubrzanja za ovu grafičku funkciju (inače vreme traženu u programima ze crtanje kao npr AutoCAD) I da ubrzanje može ići samo do približno 40 %

3 Ispis teksta rasterska operacija Obično se koriste metoda kopiranje dala video ii operativne memoriji samo što sade osim pristupa treba uradni još i neku logičku operaciju Očigledno je da je ovde dobitak manji nego kod npr brisanje memorije

A UPALIŠ MOTORKU!

Postoji doste teorija o tome kako treba vršiti testiranja komponenti računara. U konkretnom slučaju VGA kartice treba testirati specijalizovanim programom uvek u kombinaciji se pripadajućim vezmikom

U ovom slučaju su korišćeni gotovi programi *Winbench 2.51* za *Windows* okruženje i *Grperf* za *AutoCAD* (ovi drugi nisu prikazani u tekstu) Zbog sporosti ISA magistrala, danas su sve popularniji VGA grafički akceleratori Njihova prednost je brzina, a mana što za svaki program mora postojati vezmik, koji koristi specijalne mogućnosti akceleratora Bez upotrebe vezmika dobici su mali il nikakvi Na žalost razultati u dijagramu na slici 3 str 26 Računari 88 su rađeni u *Windows 3.0* okruženju. bez odgovarajućeg vezmika u razulucji 640×480 tačaka se 16 boja. Odmah se prisećamo starog vica o drvosečama kada Mujo na kraju kaže A, upališ motorku!? Tako možemo reći i D V Veselinoviću A pokreneš vezmik!?

Kad se uzmu u obzir stari rezultati i dobijeni odnosi pomoću *Winmark* testa u *Winbench 2.51* može se nacrtati korigovana tabela koja izgleda znatno drugačije Kade se izvrši korektno testiranje i upotrebi odgovarajuci vezmik u nekoj boljoj rezolucji. vidi sa da je

Nastavlja se na strani 56

BEOKURIR
GRADSKE I MEĐUGRADSKE KURIRSKE USLUGE - OD VRATA DO VRATA
Beograd
011
402 538

EKSPLOZIJA BOJA

Polako ali neizbežno, personalna informatika prelazi iz crno-belog u obojeni svet, a monitor u boji danas predstavlja standradnu opremu svake naprednije PC konfiguracije. Nakon uporednog testa monitora od 17 inča u prošlom, koji predatavljaju najbolji kompromis za korisnika *Windows*-a, u ovom broju dajemo prikaz deset modela od 14 inča, koji su namenjeni najširem mogućem tržištu.

U SREDINI: ADI Microscan 3E+

Kvalitet slike. Uočljiva greška u konvergenciji proizvodi kod ovog monitora (sa ekranskom maskom rastera 0 28 mm) neoštre ivice i to prvenstveno u srednjem delu ekrana. Utisak popravljaju dobra geometrija slike i tačan prikaz boja. Ali pošto je stabilnost slike ispod proseka i nedostaje dugme za demagnetizaciju 3E+ je ovde dobio ocenu „zadovoljava"

Veličina slike. Od celoknpna 14-inčne katodne cevi kod modela 3E+ je vidljivo 13 5 inča, te se on po tome nalazi uz bok monitoru NEC 4FG koji koristi katodnu cev od 15-inča. Na žalost korisna dijagonala iznosi samo 11 5 inča tako da je slika se površinom od 431 cm^2 ovičena širokim crnim okvirom. Zato je ocena samo "zadovoljava".

Ergonomičnost. Sa svojom maksimalnom ekranskom frekvencijom od 48 kHz *Microscan* sliku rezolucije 800x600 pnkazuje 76 puta n sakrndi. Pn rezoluci ji od 1024x768 tačake uspeva da izvuče samo 60 slika n sekundi. Slika n standardnoj VGA rezoluciji prikazuje se bez treperenja sa teoretskom maksimumom od 35 Hz. Ovaj monitor ispunjava I uslove koje postavlja MPR II norma o zračenju tako da je zaista ergonomičan. Ocena „dobar"

Podešavanje. Devet tastera koje sakriva mali poklopac ispod okvira ekrana. Imaju više estetsku nego konsnn svrhu. Pomoćn njh se regnišn vertikalna i horizontalna pozicija slike i njena veličina. Devek taster (Reset) koristi se za ponovno postavljanje fabnčki odabranih vrednosti. Na zadnjoj strani monitora nalazi se deo razervisan za servisiranje. prako koga serviser može izvršiti dodatna korekture pojedinih parametara monitora. ali sam korisnik ovom delu na može pristupiti.

Tasteri su dovoljno veliki I razmaknuti jedan od drugog tako da je isključena mogućnost neholičnog priliska na pogrešan taster. Točkući za podešavanje kontrasta I osvatljenosti kao I prakidač za ukjučenje izdvojeni sn od ostalih regnlatora izvan pokiopca I zgodno postavljeni. Pnručnik ama 12 strana i dovoljno opšimo opisnje monitor. Ocena „zadovoljavajuće"

Cena/performanse. Distributer ovog monitora je odredio cenu od 1251 DEM što je pnkladno i zaslužuje ocenu „dobar" U prodaji se ovaj monitor nudi za oko 1100 DEM. I za ovu cenu cetaje ocena „dobar"

Komforno podešavanje: Eizo T240i-M

Kvalitet slike. Tnnitron katodna cev se rastarom od 0 25 mm briljna jasnim prikazom. I greške u konvergenciji su u prihvatljivim granicama, a javljaju se li u gornjem ill n donjem delu ekrana u zavsnosti od podešavanja pomoću regulatora konvergencije. Na celom skranu nema praktično nikakvih izobličenje Zajedno se monitorom Diamond Pro 14 firme Mitsnbishi, ovaj Eizo-monitor se u pogledu geomatrija slike nalazi ispred svih ostalih testiranih modela. Nama zamerke ni na mešanje boja kod velikih površina. T240 je jedino „omanuo" kod stabilnosti slika – dimanzija slika se izraženo menjaju pri naglim izmenama crnih I belih slika. Nedostaje i Degauss-taster (za demagnetizaciju). Zato je ocene smanjena na „dobar"

Veličina slike. I T240I-M prikazuje manje nego što bi zaista mogao. Od ukupno 13 4 inča, za testiranje je preostalo samo 12 inča. 433 cm^2 donela su ocenu „zadovoljava"

Ergonomičnost. Sa velikodušno odmeranom ekranskom frekvencijom od maksimalno 60 kHz T240I M pnkazuje sliku n rezolucija 800x600 do 95 puta n sekundi. Pri rezoluciji od 1024x768 frekvencija ponavljanja slike je oko 75 Hz. Apsolutno bez treperenje je prikaz u rezoluciji 640x400 tačaka, gde bi ovaj monitor mogao prikazivati sliku 135 puta n sekundi. ali dnevnašnje grafičke kartice to još ne mogu da podrže. T240i M ispunjava i MPR II normu, tako da je u ovoj oblasti zaslužio ocenu „vrlo dobar"

Podešavanje. Eizo je ovom svom modeln podario sve kontrolne instrumente koje posednje i njegov veći rodak, 17-inčni F550i-M. Nijedna želja n vezi sa podešavanjem optimalne slika neće ostati nenaišena. Moguće je kodgovati i samn konvergencijn, kao i boje. Samo podešavanja vrši se preko jedinstvenog regulatora dok se funkcije koje se podešavajn prethodno biraju preko tastera. To je verovatno najjednostavnija i najkomfornije rešenja za ovakav posao. Moguće je memonsati 32 podešene režima. Kao I kod oba veća monitora ova firme kontrolni instrumenti su suviše pokvučeni pod sam ekran. Prateće uputstvo na 22 strana je iscrpno i pregledno. Ocena „dobar"

Cena/performanse. Ovaj Eizo-monitor uveliko prednjači po pitanjn kvaliteta slika, argonomičnosti i podešavanja. Ali preporučena cena od 2265 DEM je previsoka čak i za tako dobar 14-inčni monitor. Ocena „zadovoljavajuće". Srednja cena u prodavnicama se kreće oko 1700 DEM, što ovom kvalitetnom monitoru donosi najviši ocenu „vrlo dobar"

Tri frekvencije: Escom 2824

Kvalitet slika. Nepravilnosti n konvergenciji gotovo da I nema na skranu ovog monitora što razultuje jasnim i oštrim prikazom. ES 2824 se nzdže iznad proseka i se geomatrijom slika. Problemi se javljajn kod prikazivanja jednobojnih površina, kada se jasno vide senke. Dolazi i do razlivanja boja usled nemagnetisanosti katodna cevi. dok dugme za demagnetizaciju nije ugrađeno.

Pnlikom testiranja geometrije slika uočena su blaga odstupanja od idealno kružnog oblika kod obe sene knužnice na ekranu. Pošto nema zamerki na stabilnost slike, ocena za Escom-ov monitor je u ovoj kategoriji „dobar".

Veličina slike. Ovaj Escom-ov monitor se sa korisnom dijagonalom od 11 6 inča nalazi iza ADI-modela. Vidljivo je 13 4 inča. Za površinn prikaza od 419 cm^2 ocena je samo „dovoljan"

Ergonomičnost. Teoretski ovaj monitor (sa najvišom ekranskom frekvencijom od 48 19 kHz) prikazuje sliku 76 puta u sekundi pri rezoluciji 800x600 tačaka. Pri rezoluciji od 1024x768 tačaka, ovaj monitor uspeva da pnkaze sliku samo 60 puta n sekundi, tako slabo zrači. zadovoljava samo staru normu MPR I. Ocena „zadovoljava"

Podešavanje. ES 2824 je jedini testirani monitor koji ne radi n opsegu frekvencija već sa tri fiksirane frekvencije od 31 5 kHz, 35 5 kHz I 48 19 kHz. Zbog toga je izbor režima rada monitora moguć samo u uskim granicama. Osim toga, ovaj monitor nije radio sa grafičkom karticom konščenom na testu (Mega-Eva) i njenim standardnim parametrima. Kartica je morala biti prekonfigurisana pomoću DIP-prekidača.

I manipulsanje prikazom je svedeno na minimum. Porad osvatljenosti i kontrasta, mogu se podešavati još samo horizontalna pozicija i veličina vertikale slike. Utisak popravlja to što su točkući za podešavanje laki za okretanje i zajedno se prekidačem za uključenje postavljeni se prednja strane. Pnručnik na šest strana pruža samo najneophodnije informacije. Sve ovo zajedno zaslužnja samo ocenu „dovoljan"

Cena/performasas. U ovoj kategoriji ja monitor koga direktno nudi Escom, dobio dve ocene, jer se njegove osobina mogn sagledati sa dva aspekta. Kao prvo, se cenom od 799 DEM ovaj monitor je za 1200 DEM jeftiniji od NEC-ovog modela 4FG i za 900 DEM jeftiniji od monitora Diamond Pro 14 firme Mitsnbishi, ovaj Eizo-monitor se u pogledu geomatrija slike nalazi ispred svih ostalih testiranih modela. Nama zamerke ni na mešanje boja kod velikih površina. T240 je jedino „omanuo" kod stabilnosti slika – dimanzija slika se izraženo menjaju pri naglim izmenama crnih I belih slika. Nedostaje i Degauss-taster (za demagnetizaciju). Zato je ocene smanjena na „dobar"

Veličina slike. I T240I-M prikazuje manje nego što bi zaista mogao. Od ukupno 13 4 inča, za testiranje je preostalo samo 12 inča. 433 cm^2 donela su ocenu „zadovoljava"

Ergonomičnost. Sa velikodušno odmeranom ekranskom frekvencijom od maksimalno 60 kHz T240I M pnkazuje sliku n rezolucija 800x600 do 95 puta n sekundi. Pri rezoluciji od 1024x768 frekvencija ponavljanja slike je oko 75 Hz. Apsolutno bez treperenje je prikaz u rezoluciji 640x400 tačaka, gde bi ovaj monitor mogao prikazivati sliku 135 puta n sekundi. ali dnevnašnje grafičke kartice to još ne mogu da podrže. T240i M ispunjava i MPR II normu, tako da je u ovoj oblasti zaslužio ocenu „vrlo dobar"

Podešavanje. Eizo je ovom svom modeln podario sve kontrolne instrumente koje posednje i njegov veći rodak, 17-inčni F550i-M. Nijedna želja n vezi sa podešavanjem optimalne slika neće ostati nenaišena. Moguće je kodgovati i samn konvergencijn, kao i boje. Samo podešavanja vrši se preko jedinstvenog regulatora dok se funkcije koje se podešavajn prethodno biraju preko tastera. To je verovatno najjednostavnija i najkomfornije rešenja za ovakav posao. Moguće je memonsati 32 podešene režima. Kao I kod oba veća monitora ova firme kontrolni instrumenti su suviše pokvučeni pod sam ekran. Prateće uputstvo na 22 strana je iscrpno i pregledno. Ocena „dobar"

Cena/performanse. Ovaj Eizo-monitor uveliko prednjači po pitanjn kvaliteta slika, argonomičnosti i podešavanja. Ali preporučena cena od 2265 DEM je previsoka čak i za tako dobar 14-inčni monitor. Ocena „zadovoljavajuće". Srednja cena u prodavnicama se kreće oko 1700 DEM, što ovom kvalitetnom monitoru donosi najviši ocenu „vrlo dobar"

Tri frekvencije: Escom 2824

Kvalitet slika. Nepravilnosti n konvergenciji gotovo da I nema na skranu ovog monitora što razultuje jasnim i oštrim prikazom. ES 2824 se nzdže iznad proseka i se geomatrijom slika. Problemi se javljajn kod prikazivanja jednobojnih površina, kada se jasno vide senke. Dolazi i do razlivanja boja usled nemagnetisanosti katodna cevi, dok dugme za demagnetizaciju nije ugrađeno.

Pnlikom testiranja geometrije slika uočena su blaga odstupanja od idealno kružnog oblika kod obe sene knužnice na ekranu. Pošto nema zamerki na stabilnost slike, ocena za Escom-ov monitor je u ovoj kategoriji „dobar".

Veličina slike. Ovaj Escom-ov monitor se sa korisnom dijagonalom od 11 6 inča nalazi iza ADI-modela. Vidljivo je 13 4 inča. Za površinn prikaza od 419 cm^2 ocena je samo „dovoljan"

Ergonomičnost. Teoretski ovaj monitor (sa najvišom ekranskom frekvencijom od 48 19 kHz) prikazuje sliku 76 puta u sekundi pri rezoluciji 800x600 tačaka. Pri rezoluciji od 1024x768 tačaka, ovaj monitor uspeva da pnkaze sliku samo 60 puta n sekundi, tako slabo zrači. zadovoljava samo staru normu MPR I. Ocena „zadovoljava"

Podešavanje. ES 2824 je jedini testirani monitor koji ne radi n opsegu frekvencija već sa tri fiksirane frekvencije od 31 5 kHz, 35 5 kHz I 48 19 kHz. Zbog toga je izbor režima rada monitora moguć samo u uskim granicama. Osim toga, ovaj monitor nije radio sa grafičkom karticom konščenom na testu (Mega-Eva) i njenim standardnim parametrima. Kartica je morala biti prekonfigurisana pomoću DIP-prekidača.

I manipulsanje prikazom je svedeno na minimum. Porad osvatljenosti i kontrasta, mogu se podešavati još samo horizontalna pozicija i veličina vertikale slike. Utisak popravlja to što su točkući za podešavanje laki za okretanje i zajedno se prekidačem za uključenje postavljeni se prednja strane. Pnručnik na šest strana pruža samo najneophodnije informacije. Sve ovo zajedno zaslužnja samo ocenu „dovoljan"

Cena/performasas. U ovoj kategoriji ja monitor koga direktno nudi Escom, dobio dve ocene, jer se njegove osobina mogn sagledati sa dva aspekta. Kao prvo, se cenom od 799 DEM ovaj monitor je za 1200 DEM jeftiniji od NEC-ovog modela 4FG i za 900 DEM jeftiniji od

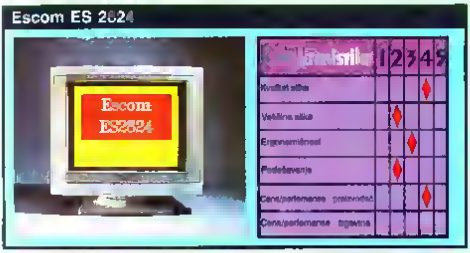
ADI Microscan 3E+

U ovim testovima se ovaj monitor se redukcranim zračenjem našao u sredini. Cena mu je sasvim povoljna.

Eizo T240i-M

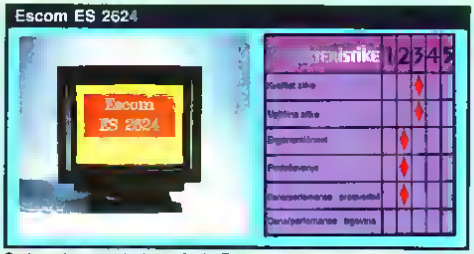
Monitor koji ce vaja komfornim podešavanjem i dobrim kvalitetom slika. Dobro odmerena prodajna cena čini ga naročito interesantnim.

Escom ES 2024



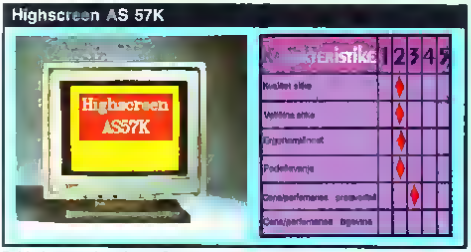
Koncepcija ovog monitora, i pored dobrog kvaliteta slike, nije ubedljiva. Na raspolaganje ima nemo tri fiksne ekranske frekvencije.

Escom ES 2624



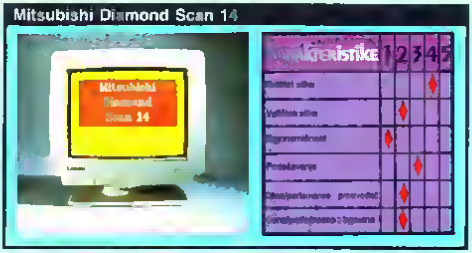
Ovaj monitor ne reducira zračenje. Zato je pri ocenjivanju odnosa cena/performanse dobio nižu ocenu.

Highscreen AS 57K



Dobio je nemo lepodprosečne ocene, a i zrači – zato mu ne pomaže ni niska cena, najniža od svih

Mitsubishi Diamond Scan 14



Niske frekvencije ponavljanja slike pri visokim rezolucijama gurnule su ovaj Mitsubishi-jev monitor u prikrajak.

nji od modela T240i-M firme Eizo. Pri tom ovaj monitor nudi kvalitetnu sliku što je neobično u toj klasi cena. Sve to zaslužuje ocenu „vrlo dobar". Sa druge strane podešavanje je lošija strana ovog monitora a naročito je zastareo koncept tada ne tri fiksirane ekranske frekvencije. Uz to ispunjava samo staru MPR I normu. Zato je za ovaj aspekt ocena „dobar".

U onom kućištu: Escom ES 2624

Kvalitet slike: Nejasne ivice i vidljive greške u konvergenciji (mešalične crvene senke pre svega n gornjem delu ekrana) umanjuju kvalitet slike. Što se toga liče, ES 2624 zaostaje za modelom ES 2824 leko je bazirao na Trinitron katodnoj cevi sa finim rasterom (0.26 mm). Sa druge strane, ovaj model deje odličan prikaz boje i ima geometriju slike koja je gotovo bez izobličenja. Stabilnost slike je na prosečnom nivou dok je dugme za demagnetizaciju uzaludno tražili. Sve zajedno zaslužuje jedno „zadovoljave".

Veličina slike: Dijagonala slike iznosi 12.1 inč dok je dijagonala vidljivog dela katodne cevi 13.9 inča 434 cm² korisne površine dobila su ocenu „zadovolja ve"

Ergonomičnost: Sa ekranskom frekvencijom od 57 kHz ES 2624 dobro stoji u ovoj kategoriji. Maksimalne nečetkenosti ponavljanja slike su 30 Hz pri rezoluciji od 800x600 tačaka i oko 70 Hz pri rezoluciji od 1024x768 tačaka. Šteta je što ovaj monitor nije „mmn" od zračenja. Zbog toga ocena je „dovoljan"

Podešavanje: Što se tiče podešavanja ovaj monitor npakovan n crno knjiške nudi malo više od svog svetlijeg brata. Mogn se podešavati vertikalna pozicija slike i njena veličine po horizontali. Dodnše ovde je položaj četin regulatora mnogo lošije izabran jer su poprilično skriveni sa leve donje strane kućišta monitora. Veliki i lako pokretljivi regulator za podešavanje osvetljenosti i kontrasta kao i prekidač za nkljnčenje monitora postavljeni su na prednje strane ispod katod ne cevi. Priručnik ne davet strana nije ništa opširniji od onog namenjenog modelu ES 2824. Ocena „dovoljan"

Cena/performanse: Cena od 998 DEM za monitor koji nije zaštićen od zračenja i koji ima prosečan kvalitet slike je dovoljna samo za ocenu „dovoljan"

Osrednji kvalitet slike: Highscreen AS 57K

Kvalitet slike: Inženjeri firme Vobis nisu baš najbolje podesili karakteristike ovom monitoru. Njegov ekran (sa rasterom tačake od 0.28 mm) pokazuje jasnn grešku u konvergenciji koja je naročito zražena u desnom delu ekrana. Zbog toga trpi i oštrina slike. *Highscreen* zaslužuje kritiku i zbog geometrije slike. Četn test-kružnice vidljivo sn se razlikovale od idealnog oblika. Donje sn bile veće nego gornje. Uz to gornje kružnice nisu bile baš savsim „okrugle". Što se ače stabilnosti slike tu su „sudije" izrekle samo dve reči „krajnje nestabilno", ali je zato razlivanje boja besprekorno. Naravno mogućnost ručne damagnetizacije ni ovde nije na raspolaganju. Za kvalitet slike: ocena je „dovoljan"

Veličina slike: U ovoj kategoriji ne *Highscreen* monitor nalazi ne samom začelju. Za upotrebu je oslavljena dijagonale od celih 11.4 inča. Ili površine od 408 cm². Vidljiva dijagonala iznosi 13.4 inča. Ocena „dovoljan"

Ergonomičnost: Kao ni ES 2624, ni Highscreen-ov malnša ne ispunjava ni jednn švedskn normu o zračenjn. Ekranska frekvencija mn je 60 kHz što znači da ne konsti ni ergonomske frekvencije koje se teorijski kreću oko 85 Hz (800x600) i oko 75 Hz (1024x768). Ocena „dovoljan"

Podešavanje: Nikakvih iznenađenja ni u ovoj kategoriji nema kod Highscreen-a. Mogućnosti podešavanja ograničene su na najnnžnije npr. veličina slike se može menjati samo po vertikali. Srećom svi regulatori sn postavljeni sa prednje strane i označeni simbolima u boji. U priručniku za npotrebu se saiče da se radi o „inteligentnom ekranu" (!?). All kada u tom istom priručniku pročitate da se ovaj monitor ne sme koristi u vlažnom prašnjavom ili suvom (!) prostoru", pomislićete da ste uzeli pogrešnu knjižicu u ruke. Ovaj priručnik je inače zadovoljavajuću ocenu srušio na „dovoljan"

KAKO IZRAČUNATI FREKVENCIJU PONAVLJANJA SLIKE

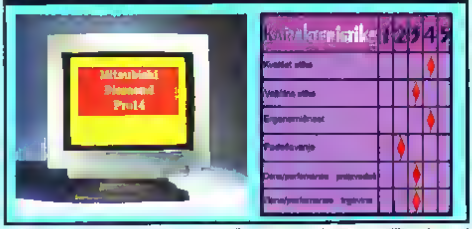
IZRAČUNAJTE I SAMI

Proizvođači uglavnom ne spominju podatke o frekvencijama ponavljanje slike svojih *multiscan* (vi šefrekvencijskih) monitore u raznim rezolucijame već radije dajn oblast ze sve frekvencije od „minimalne" do „maksimalne" (npr od 50 do 90 Hz). Uz pomoć jednostavnog računa koji međntim nije op šte poznat možete i sami proveriti koliku frekvenciju ponavljanja slike postiže neki monitor u određenoj rezoluciji. U tehničkim podacima za monitor pronađite njegovu maksimalnu ekransku frekvenciju. Na odgovarajući broj linija na ekranu dodajte još 5% da biste nzeli u obzir eventualne proizvođačke tolerancije i linije koje elektronski mlaz prelazi pri povratku na vrh ekrana i slično. Frekvenciju ponavljanje slike ze tu određenn rezolucijn sada možete izračunati ako ekransku frekvenciju podelite na brojem ekranskih linija.

Primer. Monitor ima maksimalnu ekransku frekvenciju od 60 kHz. Koja mu je najviše frekvencija ponavljanje slike pri razolnciji od 1024x768 tačaka? Račun leče ovako: 60000 podeljeno sa 806 (768 plus 5%) daje kao rezultat 74.44. Dakle monitor može pri razolucij od 1024x768 da prikazuje sliku 74 pune n sekundi e verovatno i još malo brže.

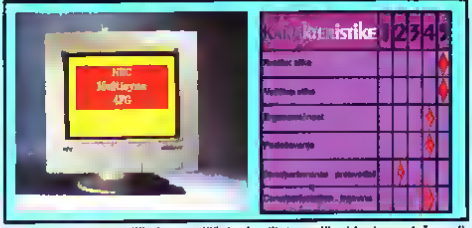
Ipak budite oprezni da li će monitori pn visokim razolucijama raditi na ovakvim frekvencijame zavisi i od grafičke kartice. Starije grafičke kartice pri razolucijame od 800x600 i 1024x768 često ne obezbeduju rad se ergonomskim frekvencijema – rade sa manje od 70 Hz. Dakle ako razmišljate o kupovini monitore ne zaodravite grafičkn karticu. Naime ukolko želite da n visokim rezolucijame radite se ekranom koji ne treperi, vervatno ćete morati da kupite novu karticu što bi mogio da bude veoma skupo. Grafičke kartice koje podržavaju visoke frekvencije ponavljanja slike (od 80 Hz pa naviše) trenutno bez izuzetka imajn vrlo visoku cenu.

Mitsubishi Diamond Pro 14



Nepraktično podešavanje je ovaj, inače dobar monitor, unazadilo pri oceni odnosa cena/performanse

NEC Multisync 4FG



15-inčni Multisync odlikuje se odličnim kvalitetom slike i brzim podešavanjima. Bio je to jedan od najboljih monitora na testu.

Cena/performanse Ovde je slična situacija kao i kod modela ES 2824 sa jedne strane, cena od 698 DEM jeste mala, ali za te pare dobijate monitor koji se u svim testiranjima našao na začelju. A i zrači. Uprkos najnižoj od svih cena, sve to je samo „zadovoljavajuće"

Jaka treperi: Mitsubishi Diamond Scan 14

Kvalitet slike Sa ovim monitorom koji koristi raster od 0.28 mm neće vam biti potrebne naočare za čitanje. Grafika i tekst se prikazuju sa oštrim vicama i uz odličan prikaz boje. Što se toga tiče, ovaj model samo neznatno zaostaje za najboljim kandidatima na ovom testu. Greška u konvergenciji je minimalna. Mitsubishi prednjači i kada su u pitanju neizobličen prikaz slika, razlašenje boje i prikaz jednobojnih oblasti. Osrednja rezolucija ovaj model ne iskazao jedino kod stabilnosti slika. Nedostaje i dugme za demagnetizaciju. Sve u svemu – ocena „dobar"

Veličina slika Ni *Diamond Scan* ne koristi u potpunosti svoj ekran. Tako slika je imala dijagonalu od 11.8 inča, što je za 1.6 inč (ili 4 cm) manje od ukupne vidljive dijagonale. 412 cm² korisne površine dostajalo je samo za ocenu „dovoljan"

Ergonomičnost Sa maksimalnom ekranskom frekvencijom od 38.5 kHz, prikaz na *Diamond Scan*-u trepan kada se radi sa rezolucijom od 800x600 tačaka (frekvencija ponavljanja slika je 60 Hz što je premalo za jedan moderan monitor). Ergonomska frekvencija ponavljanja slike od 76 Hz dobija se samo pri rezoluciji od 640x480 tačaka. Uprkos poštovanju norme MPR II ovaj monitor je ovde dobio ocenu „manjkav"

Podešavanje Kao i nekoliko drugih testiranih monitora i *Diamond Scan 14* omogućava podešavanje samo osnovnih stvari: osvetljenosti, kontrasta, veličine i pozicije slike. Šest tačkica je postavljeno sa prednje strane i omogućavaju vrlo tačno podešavanje. Uz isporučen priručnik došlo se do ocene „zadovoljava"

Cena/performanse Najveća mana monitora *Diamond Scan 14* je niska učestanost ponavljanja slike pri višim rezolucijama (60 Hz pri rezoluciji 800x600). On je zapravo upotrebljiv samo za red u standardnoj VGA rezoluciji. Stoga se cena od 999 DEM može okarakterisati jedino ocenom „dovoljan". Pošto je ovaj monitor tek stigao u prodavnice još nema značajnijih smanjenje cene

Iznad proseka: Mitsubishi Diamond Pro 14

Kvalitet slike *Diamond Pro 14* je u ovoj kategoriji ocenjen pozitivno kao i monitori *Scan 14*. Čist prikaz ne ekranu sa neznatnom greškom u konvergenciji lide ruku pod ruku sa savršenim prikazom boja. Tim više čudi nestabilnost slika pri brzim izmenama crnih i belih slika. Po geometriji slike ovaj model se nalazi iza modela *Scan 14*. Leva kružnice su ispolike. Nema ni Degauss tastera (za demagnetizaciju). Zato jedva ocena „dobar"

Veličina slika Sa 437 cm² površine namenjene slici *Diamond Pro* se nalazi ispred Scana 14 a ne nivou ADI- i Eizo monitora. Od 13.2 inča ukupne vidljive dijagonale ne raspolaganju je 12.1 inč. Ocena „zadovoljava"

ZRAČENJE MONITORA

RIZIČNO ILI NE?

Na početku jedna napomena. „Zračenje" nije tehnički ispravan naziv za različita uticaje koji izviru iz monitora. Ipak, ovaj izraz se odomaćio tako da ćemo ga i ovde jednostavnosti radi prihvatiti.

Zračenja monitora čovek niti vidi niti čuje niti ga može omirisati. Neki ljudi ga i ne osećaju dok se drugi posle dugotrajnog rada ispred monitora, žale na glavobolju, upalu očiju ili čak na promene na koži. U SAD se još uvek ne poklanja preterana pažnja monitorima sa reduciranim zračenjem, all su u Nemačkoj i Švedskoj oni već više od dve godine u žiži interesovanja. Jedno je izvesno – ili su tamo kancelarije skučenije ili su ljudi mnogo predostrožniji.

Godine 1988. Švedani su definisali granične vrednosti za „elektromagnetne emisija katodnih cevi". Švedska Organizacija radnika i službenika (TCO) izdala je tabele sa graničnim vrednostima za zračenja monitora (1987. MPR I, 1990. MPR II, 1991. TCO91 – ove tabele prilažemo).

All čak i u Švedskoj koja je najviše napredovala, sve je ostalo samo na preporukama i ne postoje striktni propisi koje proizvođači moraju poštovati. Ipak raduje činjenica da sve više proizvođača prihvata preporuke i pravi monitore koji emituju manje štetnog zračenja.

Zračenja možemo podeliti u dva kategorije: u jednoj je rendgansko zračenje (X-zraci). Ova opasnost je u skranima (računara i televizora) dobrom zaštitom svedene na minimum. Nasuprot tome vidljiva svetlost kao i električno i magnetno polje nisu zanemarljivi. U njima udele imaju optičke komponente (vidljive i ultrajubičaste) visokofrekventnog polja niskofrekvencijska magnetna i električna polja kao i elektrostatičko polje.

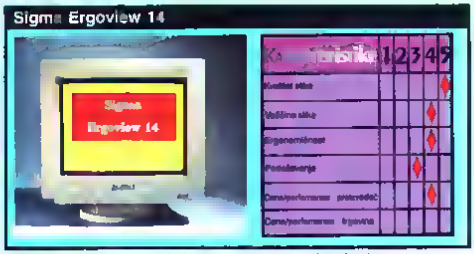
MONITORI 14 INČA – SAŽETI PODACI

	ADI Microscan 3E+	Eizo T240I-M	Escom ES 2824	Escom ES 2624
OPŠTI PODACI				
Cena proizvođača	1251 DEM	2268 DEM	798 DEM	998 DEM
Maloprodajna cena	1100 DEM	1700 DEM	****	****
EKRAN				
Veličina (dijagonala)	14 inča	14 inča	14 inča	14 inča
Vidljiva veličina (dijagonala)	13.8 inča	13.4 inča	13.4 inča	13.9 inča
Upotrebljiva veličina (800x600) (dijagonala/površina slike)	11.9 inča/431 cm²	12.0 inča/433 cm²	11.8 inča/419 cm²	12.1 inča/434 cm²
TEHNIČKI PODACI				
Tip monitora	multi	multi	3-frekvencijski	multi
Maksimalna rezolucija	1024x768	1024x768	1024x768	1024x768
Frekvencija ponavljanja slike	50 – 100 Hz	55 – 90 Hz	47 – 90 Hz	50 – 90 Hz
Ekranska frekvencija	30 – 48 kHz	30 – 60 kHz	31.5/35.5/48.19kHz	30 – 57 kHz
Opseg video signala	75 MHz	80 MHz	64 MHz	nema podataka
Raster tačaka	0.26 mm	0.25 mm	0.26 mm	0.26 mm
Norme zračenja	MPR II	MPR II	MPR I	nejasne
PODEŠAVANJE				
Osvetljenost/kontrast	da	da	da	da
Pozicija slike (vertikalna/horizontalna)	da	da	da (horizontalna)	da
Veličina slike (vertikalna/horizontalna)	da	da	da (vertikalna)	da
Konvergencija	ne	da	ne	ne
Korekcija boje	ne	ne	ne	ne
Demagnetizacija	ne	ne	ne	ne
DIMENZIJE/TEŽINA				
Dimenzije (V×Š×D)	36.2x37x38.5 cm	41x34.5x36.7 cm	34.5x37x38 cm	35x35x40.5 cm
Težina	12.5 kg	14 kg	14 kg	15 kg
GARANCIJA				
Trajanje (meseci)	12	12	6	6

Nokia 44BN-151

Prosečan monitor firme Nokia, bez velikih prednosti ili slabosti, se cenom koja je takođe prosečna.

Sigma Ergoview 14



Ergoview 14 je stvoren za pobednika, a i cena mu je pristojna.

Ugroženost od ultraljubičastog zračenja možema isključiti kao i visokofrekventno zračenje koje je prema dosadašnjim saznanjima zanemarljivo. Ne elektrostatička polja ljudi su naviknuli od rođenja.

Sasvim je drugačija situacija sa ostalim vrstama zračenja. U Švedskoj je ispitivana moguća veza između jakih ekranskih magnetnih polja i promena na koži. Ova ispitivanja su potvrdile rezultate drugih naučnih studija koje su na svetlost dana iznele povezanost kožnih oboljenja i promenljivih polja monitora, ali još nije tačno utvrđen mehanizam njihovog štetnog delovanja na ljude.

U kancelarijama se pored monitora najčešće nalaze i drugi izvori zračenja počevši od kablova i vodova za napajanje, preko neonskih sijalica, do faserskih štampača pri čemu svi daju svoj doprinos elektrostatičkom „kancelarijskom spektru". Međutim, ekrani ostaju u centru pažnje, jer su oni izvori zračenja kojima je korisnik najbliži. Jačina polja kod kolor-monitora je veća nego kod monohromatskih, pošto se kod prvih koriste tri elektronska mlaza za prikaz na ekranu, a kod drugih samo jedan.

Iako su moguće razmere bioloških oštećenja još nelistražene, svi veliki proizvođači monitora pokušavaju da kod svojih proizvoda značajno smanje rizik od zračenja. „Zdraviji bio-modeli" monitora skupi ja za otprilike 10 do 20% nego konvencionalni, što proizvođači prevdaju većim troškovima za materijal i složeniju izradu.

Jačina elektromagnetnog polja zaista se može znatno umanjiti primenom tehničkih sredstava. Međutim često će direktno na radnom mestu (dakle ispred monitora) zračenje biti smanjeno, ali će zato na drugim mestima (se zadnje strane monitora) biti povećano.

Naravno, i proizvođači dodatka se trude da popuna praznine na tržištu. Tako se nude filteri koji se stavljaju ispred ekrana, ili cela metalna kućišta u koja se stavlja monitor, ali je pitanja koliko su ova sredstva svrsishodna i koliko utiču na optimalno prilagođene komponente monitora.

I još nešto, bez obzire na sve rizike od zračenja pri izboru monitora ne treba izgubiti iz vida ni druge njegove aspekte. Oštrina slike, veran prikaz boja, prikaz bez treperenja i optimalni korisnički programi su barem isto toliko važni kada je reč o prijatnoj radnoj sredini.

Ergonomičnost: *Diamond Pro* deluje znatno profesionalnije od svog „mlađeg brata". Maksimalna ekranska frekvencija od 58 kHz dovoljna je da pri rezoluciji od 800x600 tačaka frekvencija ponavljanja slike bude oko 92 Hz. I pri rezoluciji od 1024x768 ova frekvencija je još uvek iznad 70 Hz. Skoro da se već podrazumeva da ja švedska MPR II norma o zračenju zadovoljena. Sve zajedno – ocena „dobar"

Podešavanje: Sa prednje strane monitora *Diamond Pro 14* nalaze se samo tasteri za uključenje/isključenje i regulatori za podešavanje osvetljenosti i kontrasta. Na prvi pogled sa ostale tri strane kućišta nema nikakvih dodatnih regulatora. Da nešto nije ? Ipak nije, jer su sa leve strane kućišta iza tajnog poklopca, spretno sakriveni još neki regulatori. Pomoću dva (plus/minus) tastera podešavaju se parametri koje se prethodno izaberu pomoću jednog kliznog regulatora. Mogućnosti podešavanje su zaista mnogostruke. Pored veličine slike i njena pozicija mogu se konjovati i razna izobličenja prikaza na ivicama ekrana.

Priručnik je optiman a ocena „zadovoljava"

Cena/performanse: Preporučena cena monitora Diamond Pro 14ČMDNMČ je 1806 DEM. Dobar kvalitet slike visoke frekvencije ponavljanja slike i smanjeno zračenje donose mu ovde ocenu „zadovoljava". Pošto je ovaj monitor sasvim nov na tržištu cena u prodaji se ne razlikuje mnogo od preporučene.

Nešto sasvim posebno: NEC Multisync 4FG

Kvalitet slike: Ovaj NEC-ov „maša" odlikuje se oštrim prikazom i bogatim kontrastima. Male greške u konvergenciji pojavljuju se samo u levom gornjem delu ekrana što tako i stabilnost slika i prikaz boja stavljaju monitor 4FG ispred ostalih. Jedino kod geometrije slike ovaj model pokazuje male slabosti – visina slika je desno nešto manja nego levo. Eventualna magnetizacija ekrane može se ukloniti pomoću odgovarajućeg tastera. Sva u svemu ocena „vrlo dobar"

Veličina slike: Katodnu cev ovog NEC-ovog monitora uokviruje širok ram. Zato iako je monitor opremljen 15-inčnom katodnom cevi vidljivo je samo 13,9 inča. Po tome se 4FG nalazi u istoj klasi sa ADI- monitorom. Međutim ono što ovaj monitor stvarno izdvaja od ostalih jeste optimalno iskorišćenje vidljivog dela ekrena. Za 13,6 inča korisne dijagonale (570 cm²) ocena „vrlo dobar"

Ergonomičnost: Praktično isti podaci kao za monitor Diamond Pro važe i za 4FG. Sa ekranskom frekvencijom od 57 kHz on sliku pri rezoluciji 800x600 prikazuje 90 puta u sekundi. Sa 70 Hz pri rezoluciji od 1024x768 tekođe su zadovoljeni svi zahtevi. Sve to uz poštovanje MPR II norme zaslužuje ocena „dobar"

Podešavanje: Kao i kod većeg monitora 5FG i ovde se vrlo brzo pronalaze optimalni parametri pomoću unaprst postavljenih tastera. Na raspolaganju su i mogućnosti podešavanja izobličenja prikaza na ivicama i korekcije boja. Kao i monitor firme Sigma i 4FG poseduje dugme za demagnetizaciju (koje smo već spomenuli). Priručnik na 21 strani objašnjava sve što treba da znate o instalaciji podešavanju i uklanjanju tipičnih grešaka. Ocena „dobar"

Cena/performanse: Ovaj monitor je u svim kategorijama zaslužio najbolja ocena. Zajedno sa Eizo-monitorom nalazi se ispred svih ostalih. Jeftiniji 14-inčnih monitora. Naravno, kvalitet ima i svoju cenu, ali je

Highscreen AS 57K	Mitsubishi Diamond Scan 14	Mitsubishi Diamond Pro 14	NEC Multisync 4FG	Nokia 5i	Sigma Ergoview 14
598 ****	999 DEM 999 DEM	1806 DEM 1806 DEM	2622 DEM 2000 DEM	2040 DEM 1700 DEM	1615 DEM ****
14 inča 13,4 inča 11,4 inča/408 cm²	14 inča 13,1 inča 11,5 inča/412 cm²	14 inča 13,2 inča 12,1 inča/437 cm²	15 inča 13,9 inča 13,6 inča/570 cm²	14 inča 13,2 inča 13 inča/517 cm²	14 inča 13,5 inča 12,5 inča/481 cm²
multi 1024x768 52 – 100 Hz 30 – 60 kHz 77 MHz 0,28 mm nejasne	multi 1024x768 40 – 90 Hz 30 – 38,5 kHz 40 MHz 0,28 mm MPR II	multi 1024x768 50 – 90 Hz 30 – 58 kHz 70 MHz 0,28 mm MPR II	multi 1024x768 55 – 90 Hz 27 – 57 kHz 75 MHz 0,26 mm MPR II	multi 1024x768 50 – 100 Hz 30 – 50 kHz 53 MHz 0,28 mm TCO91	multi 1280x1024 50 – 100 Hz 30 – 64 kHz 75 MHz 0,25 mm MPR II
da da da (vertikalna)	da da da	da da da	da da da	da da da	da da da
ne ne ne	ne ne ne	ne da ne	ne da da	ne ne ne	ne ne da
35,8x36,3x37,2 cm 12,5 kg	35,3x35,4x39 cm 13,5 kg	35,2x34,9x38,6 cm 13,5 kg	39,6x37,2x41,4 cm 17,5 kg	33,5x35x36,5 cm 12,5 kg	36,1x36x40,7 cm 14,5 kg
12	12	12	12	12	46

Nastavlja se na strani 24

PUNIH 10 GODINA

MICROSYS

PC386sx-33

Procesor AMD386sx 33 MHz
RAM 1 MB, SIMM
Hard disk 42 MB, IDE
Flopi 5.25', 1.2 MB
TVGA9000, 512 KB VRAM
VGA mono monitor 14"
Desktop kućište, 200W
Tastatura ASCII, 101

1399 DEM

PC386DX-40

Procesor AMD386DX 40 MHz
CACHE memorija 64 KB
RAM 2 MB, SIMM
Hard disk 42 MB, IDE
Flopi 5.25", 1.2 MB
TVGA9000, 512 KB VRAM
VGA mono monitor 14"
Desktop kućište, 200W
Tastatura ASCII 101

1668 DEM

BEZ PRAVE KO

WORKSTATION

Procesor 80c286 20 MHz
RAM 1 MB
Flopi 5.25", 1.2 MB
Herkules, TTL monitor 14"

759 DEM

NOVI SAD - Beočinski put b.b
(021) 611-366

U RAČUNARSTVU

MICROSYS

PC486DX-33

Proc. INTEL 486DX 33 MHz
CACHE memorija 64 KB
RAM 4 MB, SIMM
Hard disk 42 MB, IDE
Flopi 5.25', 1.2 MB
TVGA9000, 512 KB VRAM
VGA mono monitor 14"
Desktop kućište, 200W
Tastatura ASCII 101

2657 DEM

PC486DX-50

Proc. INTEL 486DX 50 MHz
CACHE 8 KB + 256 KB
RAM 4 MB, SIMM
Hard disk 42 MB, IDE
Flopi 5.25', 1.2 MB
TVGA9000, 512 KB VRAM
VGA mono monitor 14"
Desktop kućište, 200W
Tastatura ASCII 101

3365 DEM

486DX2-66

Proc. INTEL 486DX2 66 MHz
CACHE 8 + 256 KB
RAM 4 MB, SIMM
Hard disk 42 MB, IDE
Flopi 5.25', 1.2 MB
TVGA9000, 512 KB VRAM
VGA mono monitor 14"
Desktop kućište 200W
Tastatura ASCII 101

3744 DEM

NKURENCIJE !

DOPLATE

SVGA kolor monitor 14" 1024x768	292 DEM
1 MB RAM	78 DEM
FDD 3.5"	135 DEM
Disk 89 MB	139 DEM
Disk 105 MB	207 DEM
Disk 125 MB	277 DEM
Disk 205 MB	605 DEM

ŠTAMPAČI

HP IIIp - 1 MB, toner
HP IV - 2 MB, toner

EPSON LX-400, YU-set
EPSON FX-1050, YU-set
EPSON LQ-570, YU-set
EPSON LQ-1070, YU-set
EPSON DFX-5000

M R E Ž E

Novell NetWare 2.20
Novell NetWare 3.11
Novell NetWare Lite
SCO TCP / IP
Ethernet card NE2000 kom
ArcNet card 8/16 bit
ArcNet Active Hub 4/8
Pribor, Instalacija, Podrška

BEOGRAD - Molerova 70
(011) 432-690, 430-059

SERVER PO KRVI

Svedoci smo sve bržeg porasta znsčaja I primene računarskih mreža, a prvi korak obično predstavlja neka aktivnost u vezi sa lokalnim računarakim mrežama (LAN). Cilj ovog članka je da predstavi pregled funkcionalnih karakteristika PC kompatibilnih servara i naka aktuelna tehnička rašenja. Predstavićemo COMPAQ SYSTEMPRO/LT kao jedno konkretno rašenja i prikazati deo njegovih mogučnosti u ulozi servera mreže, poredeći ga sa „klon" serverom.

Any to any using many

Spiridon Matić

Diskusija o performansama I konkretnim ciframa (msec, KBPS i slično) je prilično nezahvalan i jalov posao, jer je za korisnika ipak najvažnije da sistem efikasno (pouzdano, itd) izvršava konkretnu klasu poslova. Sa druge strane, postavlje se pitanje izbora test-programa I radnog okruženje, tako da je prava vrednost svakog rezultata testiranja uvek prilično neizvesna. Svih ovih opasnosti smo bili svesni i ovom prilkom, pa smo se odlučili za sledeći pristup. Na dve konkretne mreže – jedna sa COMPAQ-ovim serverom a druga sa klonom (ovaj drugi je, dakle, server samo po funkciji), oba servera sa identičnim nazivnim performansama ključnih elemenata (vrsta i takt procesora, veličina I brzina memorije, kvalitet i kapacitet podsistema diskova) – isprobaćemo nekoliko tipičnih aplikacija iz baza podataka i uporedili rezultate, koristeći *Novell NetWare 3.11*. Ne pretendujemo da iz svega izvlačimo dalekosežne zaključke, ali smatramo da će barem jedna grupa korisnika imati koristi od ovakvih iskustava I da će sagledati razlike među mrežama sa „serverom po krvi" I „serverom funkcij".

Snimio: Studio Tešić & Nenad Petrović

OPŠTA ANATOMIJA

Prvi korak u definisanju i uspostavljanju tržišta servera učinila je firma COMPAQ COMPUTER CORP. predstavljanjem svoje mašine SYSTEMPRO, krajem 1989 godine. Na taj način je i formalno predstavljena nova klasa mašina, tzv. nemenskih servera (*dedicated file server*), koji zadovoljavaju potrebe velikih računarskih mreža, kombinujući snažne komponente, obezbeđujući prošrivost i pouzdanost (mi, nažalost, jednako prevodimo *dependability* kao i *reliability* – ovde podrazumevam pouzdanost u smislu prvog pojma)

U proteklom periodu veći broj proizvođača računara je predstavio svoje servere. Obično je reč o najjačim mašinama iz proizvodnog programa odgovarajućeg proizvođača. U redim slučajevima reč je o mašinama koje su posebno projektovane za ove namene (COMPAQ, IBM, Tricord, Everex, Dell, ...), a u još redim radi se o tzv. „superserverima", koji mogu imati do 10 procesora (na primer, NF450FT firme „Netframe Systems")

Server je nova vrsta računam projektovana da opslužuje korsnika („klijenta") u mrežnom okruženju. Dakle, server je tipičan višekorisnički resurs, koji stoji na raspolaganju korisnicima u mreži. Postoji nekoliko elemenata značajnih za ispravno i efikasno funkcionisanje mreže, ali većina svih zahtava koje generišu korsnici, kao i većina podataka koji kruža kroz mrežu, pre ili kasnje završava u serveru, odnosno na njegovom disku. Najveći teret mreže je na plećima servera i stoga on mora biti kombinovan od najboljih elemenata PC mašina, mniračunara, inženjerskih radnih stanica i velikih računara.

Jedna moguća klasifikacija okruženja servera bazirana je na broju korsnika, radna grupa, odeljenje, cela kompanija. Nisu usvojene konkretne cifre za razgraničenje ovih klasa, ali je sadašnja nomenklatura COMPAQ-a sledeća do 50, do 200, preko 200 korsnika.

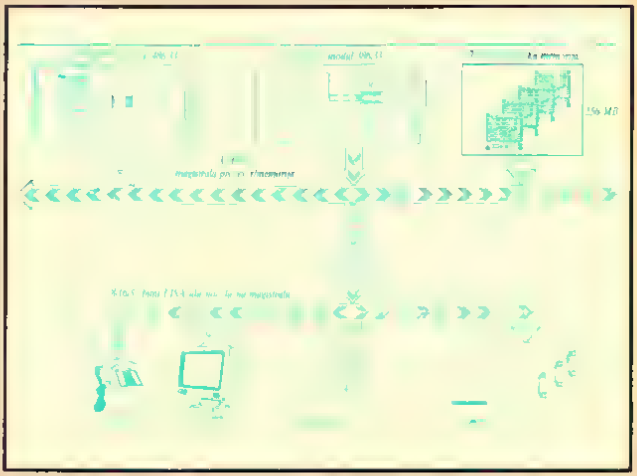
U pogledu klasifikacije poslova, takođe postoji veliko šarenilo. Predstavljamo jednu moguću nomenklaturu:

- **File/Print usluge**: centralizovano čuvanje zajedničkih podataka, štampanje, pristupa veći broj korisnika koji ne radnim stanicama izvršavaju prevashodno jednokorsničke aplikacije (obrada teksta, tabelama izračunavanja i slično)
- **Komunikacije:** pristup drugim mrežama, konsteći zajednički *bridge* ili *gateway*, kao i korišćenje elektronske pošte unutar lokalne mreže
- **Aplikacioni programi:** automatizacija specifičnih funkcija radne grupe konsteći DBM (*Data Base Management*) sistam
- **Kombinovane usluge:** kombinacija dve ili više gore pomenutih usluga na jednom serveru

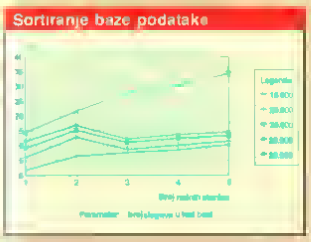
OSNOVNE KARAKTERISTIKE

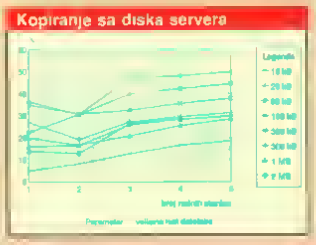
Postoje različita mišljenja o značaju i prioritetu pojedinih karakteristika servera, sa stanovišta korsnika. Jedna moguća i često citirana klasifikacija bi bila sledeća

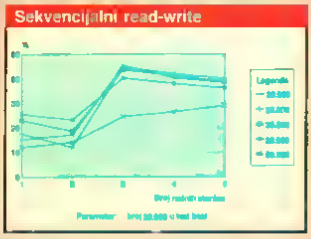
- kompatibilnost,
- pouzdanost,
- mogućnost nadgradnje (*upgradeability*),
- mogućnost proširivanja (*expandability*),
- performanse,
- dobavljivost.

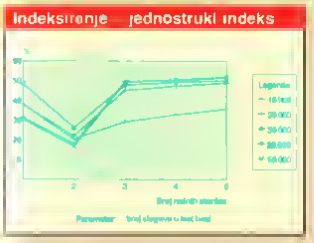


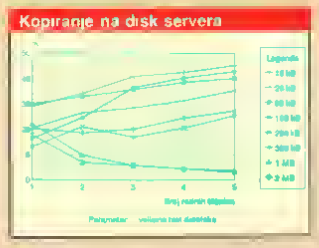
Slika 1 Flex/MP arhitektura

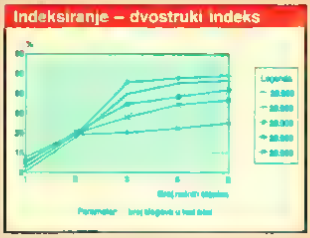












Kompatibilnost je značajna karakteristika sa aspekta realizacije mreža, e omogućava konačenje proizvoda (hardverskih i softverskskih) različitih proizvođača (*multivendor solutions*), ili pak proizvoda istog proizvođača, ali iz različitih vremenskih perioda. Iskustvo nas uči da po pitanju deklarisane kompatibilnosti proizvoda treba biti veoma oprezan jer je složenost današnjih proizvoda (kako hardverskih tako i softverskih) takva da se neki latentni uzroci mogućih problema otkrivaju posle dužeg ili kraćeg perioda eksploatacije („sve je to kompatibilno samo je nešto malo manje a nešto malo više kompatibilno")

Bez obzira što je server samo jedan od elemeneta mreže, ima vrlo mnogo argumenata u prilog tezi da njegova **pouzdanost** postaje jedne od najznačajnijih karakteristika. Poboljšanju pouzdanosti pristupa se na dva načina

Prvi je poznat pod nazivom *fault avoidence* i predstavlja skup metoda, postupaka i tehnologija kojima se preventivno deluje na pojavu eventualnih grešaka. Najkraće rečeno, reč je o projektovanju i previjenju kvalitetnog proizvoda.

Drugi način poznat je pod nazivom *fault tolerance* i predstavlja skup metoda postupaka i tehnologija kojima se obezbeđuje da pojava greške ili otkaze u serveru ne dovede do zastoja u redu celog sistema. Pri tome se, naravno, dozvoljava da se performanse sistema degradiraju do određenog stepena. Od poznatih tehnika koje obezbeđuju otpornost na otkaze u današnjim serverima se tipično primenjuje zaštita operativne memorije (RAM), što već i nije specifičnost samo servere, kao i zaštita podsisteme diskova, primenom različitih nivoa RAID tehnike. Ovde bi se još mogle svrstati i primene upravljivih izvora neprekidnog napajanje (UPS)

Mogućnost nadgradnje predstavlja svojstvo koje obezbeđuje očuvanje prvobitne investicije korisnika i povećanje performansi servera, zamenom nekih ključnih elemeneta novijim, kao što je npr zamene CPU (Central Processing Unit) ploče novijom, bržom COMPAQ je uveo ovakvu praksu za svoje mašine bazirane na EISA magistrali, pri čemu korisnik plaća samo razliku u ceni

Mogućnost proširivanja predstavlja svojstvo koje obezbeđuje očuvanje prvobitne investicije korisnika i povećanje kapaciteta pojedinih resursa servera (RAM, diskovi, povećanje broja procesora u višeprocesorskom serveru i slično).

Posmatrano sa tehničkog stanovišta, krive **performansi** mreže, bez obzire kako ih definisali i menjali, bez obzire na tip servera i klasu poslova, pokazuje tendenciju zasićenja, što je uostalom karakteristika svih tehničkih sistema. Na početnom delu krive, „kapacitet" servera je nepotpuno iskorišćen – dakle, mogu se dodavati novi klijenti, bez uticaja na ukupne performanse. Nakon zasićenja servera, ukupan protok podataka kroz mrežu ostaje približno konstantan, dok svaki klijent ponaosob očekuje njegov pad (linearan ili još i brži), sa daljim povećanjem ukupnog broja klijenata.

Nivo zasićenje mreže zavisi od više različitih faktora. Ovde ćemo nevesti one koji se tiču samog servere: podsistem CPU/memorija, podsistem diskova, podsistem mrežnog interfejsa.

Pod pojmom **dobavljivost** podrazumevamo više različitih parametara, koje svaki potencijalni i ozbiljni konsnik more uzeti u obzir pri opredeljivanju za konkretni proizvod: da li postoji ovlašćeni zastupnik ili distributer, tj. mogućnost nabavke od „lokalnog" prodevca, uslovi i procena realnih mogućnosti realizacije ponuđenih uslova garancije, servisiranje' i podrške u garantnom i vangarantnom roku i, konačno, nabavna cena. Naravno, svaki korisnik će, na bazi ovih parametara i eksploatacionih parametara celog budućeg sistema (koliko košta zastoj sistema, itd.) definisati svoju politiku.

TEHNOLOGIJA SERVERA

Današnji serveri tipično koriste jedan ili više procesora (najčešće dva). U ovom drugom slučaju (naspr. ALR, COMPAQ, EVEREX ...), obično je primenjen princip simetrične arhitekture. Dakle, oba procesora su ravnopravna i izvršavaju poslove koje im odgovarajući dispečer (*scheduler*) dodeljuje. Za razliku od ovoga, postoji i asimetrični slučaj, gde su procesori specijalizovani po funkcijama koje obavljaju. Upotrebljene je brze magistrala za prenos operativnoj memoriji, uz intenzivnu primenu *cache* tehnika. Tipični minimalni kapacitet operativne memorije je 8-16 MB, dok je maksimalno mogući u najvećem broju slučajeve 64 MB (COMPAQ SYSTEMPRO do 256 MB). Za veličinu spoljašnje „keš" memorije uglavnom se usvajaju vrednosti između 128 kB (najčešće) i 512 kB, dok u izboru arhitekture i algoritma „keširanja" postoji veliko šarenilo *two-way set associative, write through, direct-mapped, write back, direct-mapped, write through* – da pomenemo samo neke od češće primenjivanih, ali o detaljima, možda, nekom drugom prilikom. Teorijski i simulacioni rezultati o pogledu performansi i nedostataka pojedinih algoritama su dobro poznati, ali bi bilo zanimljivo ispitati njihov uticaj na performanse realnog sistema za pojedine tipične klase aplikacija.

U pogledu izbora I/O (input/output) magistrale, opredeljenje većine proizvođača je za EISA arhitekturu, dok je znatno manje za MCA. Naravno, još uvek ima i rešenja sa ISA magistralom. Broj slobodnih utičnica na magistrali kreće se u od 6-10, u zavisnosti od proizvođača.

Problem podsistema diskova svodi se na pitanje izbora sprežnih kontrolera, organizacije podsisteme i kapaciteta. Naravno, zahtevi korsnika su što veći kapacitet, što veći protok i što veća pouzdanost. U primeni je nekoliko različitih tipove sprežnih kontrolera IDE, ESDI, SCSI, SCSI-2, IDA, IDA-2 IDA i IDA-2 su isključivo 32-bitni EISA kontrolen, dok se ostali tipovi mogu naći u 16- i 32-bitnoj verziji. Najčešće se primenjuju inteligentni kontrolen, koji obezbeđuju nekoliko različitih efekata.

Tehnikom „keširanja" i primenom nizova diskova omogućuje se veći protok. Kapacitet „keš" memorija disk kontrolere tipično se kreće u granicame 1 – 4 MB, dok npr. *SCSI Driver*

Array firme „FCS Ltd." može imati do 64 MB. Nizovi diskova predstavljaju novi pristup u organizaciji podsistema diskova koji, porad povećanog protoka i većeg kapaciteta, omogućava realizaciju pouzdanih podsistema diskova, primenom RAID tehnike (vidi COMPAQ Bilten u ovom broju). Izbor konkretnog tipa organizacije je stvar kompromisa između traženog raspoloživog kapaciteta (dakle, stvarni, fizički umanjen za deo koji se koristi za povećanje pouzdanosti), protoka (koji, opet, zavisi i od klase aplikacija) pouzdanosti i, naravno, cene.

Tipični kapaciteti diskova kreću se u opsegu 1-4 GB.

COMPAQ-ov PRISTUP

Unapređujući opšte usvojenu osnovnu koncepciju konkurentne magistrale, naročito primenom posebnih metoda za „keširanje", COMPAQ je zaokružio svoju arhitekturu, poznatu pod nazivom Flex. Kasnije je razvijena i proširena verzija, Flex/MP, koje je predviđena za višeprocesorski rad. Principijelna šema Flex/MP arhitekture prikazana je na slici 1.

EISA specifikacija predstavlja proširenje rešenja usvojene ISA (Industrial Standard Architecture) magistrale. Predviđena je za snažnije sisteme, što se vidi iz osnovnih karakteristika:

- podržava 32-bitne *bas mastere* (do 15 mastera),
- podržava 32-bitni DMA rad na magistrali.

Sa druge strane, obezbeđena je potpuna kompatibilnost sa ISA magistralom, što omogućava korišćenje svih postojećih dodatnih modula i kontrolera.

Kada su jedinice diskova u pitanju, COMPAQ podržava većinu sada postojećih standarda (IDE ESDI napredni ESDI (sa EISA spregom), SCSI-2, IDA i IDA-2). Ovom prilikom skrećemo pažnju na IDA i IDA-2 kontrolera, koji podržavaju nizove diskova, obezbeđujući, pored povećana pouzdanosti i bolje performanse.

COMPAQ SYSTEMPRO/LT je prikazan sa više aspekata u dosadašnjim izdanjima COMPAQ Biltena. Jedno značajno svojstvo familije COMPAQ SYSTEMPRO prikazano je u ovom Biltenu. Zbog svega ovoga, u ovom tekstu smo se ograničili na prikazivanje nekih karakteristočnih tehničkih rešenja i na testiranje performansi SYSTEMPRO/LT servera u *Novell NetWare* okruženju.

Testirana standardna konfiguracija spada u kategoriju najslabijih COMPAQ-ovih servera: procesor 386/33, RAM 8MB, 210 MB IDE disk, 16 kB *four way set-associative* keš memorija.

OČI U OČI

Kion mašina za poređenje je imala sledeću konfiguraciju: procesor 386/33, RAM 8MB, 210 MB IDE disk, 64 kB keš memorije.

Kao radne stanice upotrebljene su 286 mašine, sa procesorom na 16MHz, RAM 2 MB i sa 44 MB IDE diskovima.

Očigledno je da bi, ako bismo ovako uporedivali oba servera, dobijeni rezultati, barem što se tiče performansi, bili približno jednaki, i tu na bi trebalo ništa posebno ni ispitivati – zbog toga što, u tom slučaju, ni jedan od resursa COMPAQ mašine, koji je kvalitetno razlikuje od klona, nije iskorišćen. Obe mašine su pre podizanje mreže testirane programom BENCH, ver. 6.0 (PC MAGAZINE). Dobijeni su vrlo slični rezukati, sem u pogledu video podsistema (COMPAQ je bio drastično brži, ali ovo nije od značaja za njegovu funkciju mrežnog servera).

Jedna od značajnih razlika je to što COMPAQ ima 32-bitnu EISA magistralu, tako da je omogućeno korišćenje brzih periferijskih kontrolera. Da bismo donekle uspostavili realnije odnose, COMPAQ je testiran se 16-bitnom mrežnom karticom, dok je klon server imao 8-bitnu karticu istog proizvođača. Tako je svaki server bio za korak iza najboljih mogućih uslova, u pogledu mrežnog interfejsa. Smatramo da rezultati testiranja ovako organizovanog okruženja imaju određeni praktični značaj, jer prikazujemo samo relativna odnose rezultata merenja performansi oba servera.

Sve radne stanice su bile opremljene 8-bitnim mrežnim karticama i instalirane ja mrežni softver *Novell NetWare ver 3.11*.

Formirana je mreža sa 5 radnih stanica. Na svakoj radnoj stanici se izvršavao identičan program, pri čemu je broj radnih stanica variran od 1 do 5. *FoxProLAN* je konfigurisan tako da su svi radni katalozi smešteni na disk servera. Dakle, maksimalno je opterećen disk servera i sama mreža.

Testovi merenje brzine rada se bazom podataka rađeni su na bazama veličine 10 000, 20 000, 30 000, 40 000 i 50 000 zapisa. Test programi su pisani u jeziku *FoxProLAN* verzije 1.02. Korišćeni su sledeći testovi:

- READ: sekvencijalno čitanje jednog polje iz zapisa, modifikacija i ponovni upis,
- SORT: sortiranje baze,
- INDEX1: indeksiranje baza po jednom jednostrukom indeksu,
- INDEX2: indeksiranje baze po jednom dvostrukom indeksu.

Za merenja brzine rada se datotekama korišćene su datoteke veličine 10 kB, 20 kB, 30 kB, 100 kB, 200 kB, 300 kB, 1MB i 2MB.

- COPY1: kopiranje datoteke sa diske na disk servera. Izvodi se 10 puta kopiranja i brisanje datoteke,
- COPY2: kopiranje datoteke sa diska servera na disk radne stanice. Izvodi sa 10 puta kopiranja i brisanja datoteke.

REZULTATI TESTIRANJA...

... predstavljenih konfiguracija mogu se sumirati u dve grupe utisaka. Prvu grupu je teže kvantifikovati, a sastavljena je od ocena i iskustava dobijenih, pre svega, u periodu intenzivnog rada sa konfiguracijom. Najkraće rečeno, COMPAQ ja pouzdana mašine sva instaliranja softvera izvode se pravolinijski, bez ikakvih nesugodnih iznenađenja a ponašanje mašina u eksploataciji je stabilno.

Druge grupa rezultata predstavljena je grafički. Na slikama su prikazani rezultati merenja performensi primenom test-programa READ, SORT, INDEX1, INDEX2, COPY1 i COPY2, respektivno. Prikazano je relativno povećanje vremana izvršavanja pojedinih programa na klon serveru u odnosu na COMPAQ server.

COMPAQ SYSTEMPRO/LT se, naravno, može koristiti i baz IDA podsistema diskova i 32-bitna mrežne kartice, ali u tom slučaju se neće osetiti njegova puna snaga i korisniku će biti uskraćena mnoga zadovoljstva. Zadovoljstvo rada sa kvalitetnom pouzdanom i stabilnom mašinom je uvek vrlo prsutno, čak i kad vozite sjajan automobil konsteći samo prvu i drugu brzinu. Ali, za puni doživljaj treba još nešto više. O tome u jednom od naših sledećih priloga.

MONITORI OD 14 INCA

Nastavak sa strane 19

preporučena cena od 2622 DEM ipak previsoka, čak i za monitor koji se zasniva na 15-inčnoj katodnoj cevi. Tu je ocena samo „dovoljan". Srednja prodajna cena kreće se oko 2000 DEM, što već zaslužuje ocenu „dobar".

Pomalo preskup: Nokia 448N-151

Kvalitet slike: Zaista oštar prikaz na ekranu ovog monitora kvan grešaka u konvergenciji u donjem delu ekrana, koje se manifestuje crvenkastim senkama. Zbog toga su linije i ivice u toj zoni blago nejasne. Geometrija slike je prihvatljiva (oba lava kruga su malo jajolika). Prikaz boja i stabilnost slike su bez greške. Nedostaja mogućnost ručne demagnetizacije ekrana. Ocena „zadovoljava".

Veličina slike: Monitor firme Nokia se u ovoj kategoriji našao na drugom mestu, iza NEC-ovog monitora. Od 13,2 inča ukupne vidljive dijagonale iskorišćeno je 13 inča, što daje korisnu površinu od 517 cm² i za diaku ocenu „vrlo dobar".

Ergonomičnost: Dok pri rezoluciji od 800x600 tačaka ovaj monitor se ekranskom frekvencijom od 50 kHz ima frekvenciju ponavljanja slike od dobrih 80 Hz, dotle ona na rezoluciji od 1024x768 pada na nekih 60 Hz. All zato ovaj monitor, jedini od svih testiranih zadovoljava uslove koje postavlje nova, strožija norma o zračenju – TCO91. Ovu poslednje je uticalo da monitor u ovoj kategoriji dobije ocenu „dobar".

Podešavanje: Svi elementi pomoću kojih se vrše podešavanja, zajedno sa prekidačem za uključenje/isključenja monitora, ugrađeni su u desnu stranu kućišta. Pored osvetljenosti i kontrasta, sa četiri potenciometra, točkića se regulišu i veličina slike i njena pozicija. Priručnik na sedam strana objašnjava samo ono najosnovnije. Sve to zaslužuja samo ocenu „dovoljan".

Cena/performanse: Preporučena cena da 2040 DEM je za ponuđenu kvalitet slike i nepraktično podešavanje previsoka – uprkos poštovanju stroge TCO91 norme. Ocena „dovoljan". Ocenu „zadovoljave" dobija srednja prodajna cena ovog monitora, koja iznosi oko 1700 DEM.

Nov i kvalitetan: Sigma Ergoview 14

Kvalitet slika: Sa svojom Trinitron katodnom cevi se rastarom od samo 0,25 mm. *Ergoview* je imao najbolje predispozicije da u ovom delu testa zauzme neko od prvih mesta i on je to očekivanje i ispunio. Prikaz na ekranu je veoma čistar, tako da su čak i pri višim rezolucijama, i mala slova lako čitljiva. Na gornjoj ivici slike primećuje se mala greška u konvergenciji. Oristupanje kod geometrije slike i stabilnosti takođa su neznatna. Raduja činjenica da je na raspolaganju i dugma za demagnetizaciju, tako da oceni „vrlo dobar" u ovoj kategoriji ništa nije stajelo na putu.

Veličina slike: Ovde monitor nije dobio najvišu ocenu, ali nije ni loše prošao. Za korisnu površinu ekrana od 481 cm² (12,5 inča od 13,6 inča vidljive dijagonale) *Ergoview 14* je dobio ocenu „dobar".

Ergonomičnost: Odličnih 64 kHz ekranske frekvencije omogućavaju frekvenciju ponavljanja slike od 100 Hz pri rezoluciji od 800x600 tačaka, što je tek poneka grafička kartica u stanju da prati. I pri rezoluciji od 1024x768 tačaka ova frekvencija još uvek iznosi celih 80 Hz. Takođe *Sigma Ergoview* poštuje MPR II normu. Ocena „dobar".

Podešavanje: *Sigma Engoview 14* nudi ručnu demagnetizaciju i regulisanje izobličenja na rubovima ekrana. Oba ova regulatora smeštena su se zadnje strane monitora. Sa prednje strane pronaći ćete preostale regulatore za podešavanja osvetljenosti kontrasta, veličina i pozicija slike. Veliki odrtni regulatori omogućavaju brzo i lačno podešavanje.

Pratiće uputstvo ne 16 strana je iscrpno napisano. Sve u svemu – ocena „zadovoljava".

Cena/performanse: Za 1815 DEM koilka mu je preporučena cena, *Ergoview 14* nudi najbolji kvalitet slika visoke frekvencije ponavljanja slike i zaštitu od zračenja. To je za ocenu „dobar". Ovaj model je sasvim nov, ali se može očekivati da prodajna cena bude oko 10-20% ispod preporučene. Tada će Ergoview po odnosu cena/performanse daleko odmaći od konkurencije.

Sa nemačkog preveo
Kragi Aleksandar

A new Star

You can finally reach the star

Star ComputerPrinter LC-20

star MICRONICS
the ComputerPrinter

AUTHORIZED DEALER "SAGA" D.O.O. BEOGRAD, YUGOSLAVIA
Milentija Popovića 1 "SAVA CENTAR" 11070 NOVI BEOGRAD
Tel: & Fax: 011/222-3579, 147-182, 222-4323 ext. 256 & 259

POVUCITE PRAVI POTEZ

Potencijalnom kupcu DOS-PC računara može se završiti u glavi od šarolika ponude mikroprocesora. Četiri kompanija – Intel, AMD, Cyrix i IBM – nuda najmanje 14 varijanti procesora iz klasa 386 i 486, sa brzinama od 16 do 66 MHz. Samo Intal nudi preko 100 varijanti mikroprocesora 80x86, dok za 1993. planira 25 varzija 486-tica. Ne zaboravimo ni uskoro očakivani Intalov čip P5, kao ni nova klonove čipa 80x86, koje za prvu polovinu godina obećavaju ostali proizvođači. Trka za najbrži mikroprocesor uneia ja na tržišta konfuziju, ali i donosi niža cene PC sistama.

Izbor se više ne može zasnivati ni na ceni ni na performansama. Potrošača struja pnsustvo malarna tičkog procesora i slepen integrisanosti u velkoj meri odražuju kako će se CPU nositi s vašim dragocenim aplikacijama. Zar je onda čudno što je kupac PC računara zbunjen?

Najzbunjeniji su individualni korisnici i oni u malim firmama. „Da li je 386/40 brži od 486SX/20? Takve odluke su tvrd orah čak i za profesionalne prodavce računara. Poseban predmet zabune je razlika izmedu SX i DX klasa čipove. Jedan iskusni prodavac kaže da ni ne pokušava da sve to razjasni svojim mušterijama – samo se trudi da im ponudi sistem koji će najbolje odgovoriti njihovim potrebama.

Proceal ponude mikroprocesora dobnim delom je rezultat utrkivanja ave većeg broja proizvođača za mesto na tržištu ali ja ta raznolikost i odgovor na istinska potrebe korisnike. Ne primei novi Intelov procesci 486SL koji troši malo struje je prevashodno namenjen notebook računarima, gde su ranija dugotrajnost baterija visoka performansa bill pojmovi koji su se uzajamno isključivali. Cyrix-ov procesci DRu2 koji uskoro treba da se pojavi će se prodavati kao dilaktne dogradnja (nožica na nožicu) iz klasa 486SX za korisnika Intelovih 386DX mašina

Da bi se napravio pravi izbor sistema neophodno je poznavanje sknvenih jakih i slabih strana kako postojećih tako i tak najavljenih procesora. Potrebno je znati i odnos cena i performansi kao i koji CPU nudi najbolje mogucnosti dogradnje

Srećom pitanje kompatibilnosti se ne postavlje. Magazin BYTE je intervjusao brojne izvore i sproveo niz testove sa čipovime Intela AMD a i Cyrix a. Niko nije pomenuo problem kompatibilnosti. Predsednik kompanija „Acer America" Ronald Čvang (Ronald Chwang) je izjavio da je AMD 100% kompatibilan sa Intelom. Po rečima Džona Patersona (John Patterson) iz Tandy ja „kompatibilnost je apsolut" Tandy ima potpuno povorenje u Cyrix-ov 486SLC i koristi ga za svoj 3800 HD notebook PC. Najozbiljnnji problem uočen u našim testiranjima je nesposobnosti nekih stanjh programa da prepoznaju novije mikroprocesore

POKRETAČKE SILE

Konkurencija je ubrzala razvoj mikroprocesora. Dok je Intel držao celokupno tržište pocesora 80x86 mogao je sebi dozvoliti duže izazvojne cikluse. Sada AMD i Cyrix proizvode procesore koji konkurišu postojecim Intelovim jednicama ili nuda bolja performanse i to često po nižoj ceni

I IBM proizvodi unapređene verzije procesora 386SX i 486SX koristeći licencu koju je otkupio od Intela. Po tom ugovoru IBM ima pravo da ih koristi za svoje sistema ili da prodaje matične ploče sa tim čipovima

Takve konkurencija je stvenia kupovno tržište za proizvođače PC ja i doprinela sniženju cena sistema. Jedan amančki proizvođač matičnih ploča kaže da je AMD kao konkurent natjerao Intel da iz temelje izmeni cenovnik 386 familije čipove. Slika 1 pnkazuje utcaj konkurencija na Intelov cenovnik. Cena 16-MHz 386DX procesora je pala za oko 20% od kraja 1968 do drugog kvartale 1991 godine kada je AMD izbacio svoj 386DXL. Od tada do danas Intel je smanjio cenu 386DX za 43 posto za trostruko kraće vreme.

Projektovanja mikroprocesora koji će biti poslednje reč tehnologija nje ni malo lak posao. Cyrix je potrošio 10 miliona dolara da bi izneo 486SLC i 486DLC na tržište što je tričave sume u odnosu na 250 miliona koje je uložia Intel u razvoj svog procesora 486. Medutim veličina mikroprocesorskog tržišta dalje primamljivo „Dalaquest" firma koja se bavi istraživanjem tržišta procenjuje da je celokupno CPU tržiste (uključujući i čipove koji ne spadaju u klasu 80x86) u periodu od 1990 1991 vrsdelo gotovo 4 milijarde dolara od čega je 70% pripadalo Intelu i AMD-u i novi proizvodi odnosa veliki deo kolaća, kako u malim tako i u velikim tržišnim segmenlima u što se AMD uvano

AMD je izbacio prvi klon Intelovog procesora 386 nakon duge sudske bike sa Intelom. Presuda je odlučila da AMD ima pravo da prodaje svoj CPU na osnovu ugovora koji je potpisan u vreme kada je procesor 286 bio na vrhuncu. Od tada je AMD veoma uspešan na tržištu 386-tice – kompanija je planirala da do kraja 1992 proda preko 2 miliona mikroprocesoie iz klase 386 što čini gotovo 50% udele ne tržištu

AMD je predstavio sistem sa 50 MHz 486DX čipom koji koristi zaštićeni Intelov mikrokod. Predsednik kompanije Džen Sanders (Jerry Sanders) izjavljuje da će početkom 1993 izbaciti ovaj CPU na tržišta bilo kako šta će dobiti pravo na korišlenje mikrokoda sudskim putem ili tako što će napisati novi mikrokod

Diversifikacija PC sistema stvonia je prilku za nove projekte mikroprocesora. Nekadašnja filozofija „jedan čip za sve nlamene" zamenjene je procesornima projektovenim preme specifičnim potrebame 386 i 486 procesori iz klasa SL koj troše malo struje konsta se gotovo isključivo za prenosive (portable) sisteme koji rade na baterije. Jeftini procesori iz klase SX postali su uobičajene za ulazne desktop sisteme. Klasa 486DX se visokim performansama opslužuje high-end radne stanice i servere. A u poslednje vreme visoko integrisani CPU iz klasa 8086 i 286 sve su popularniji izbor za novu kategoriju „ručnih" (hand held) računara (vidi leksi „Rezervni igrači osvajaju nove segmante tr žišta)

BYTE je razgovarao sa 20 vodečih svetskih proi-

LABORATORIJSKI TEST: DA LI JE VAŽNA MARKA?

Iako je količina i raznolikost procesora koji se nude pomalo zbunjujuća, pravo pitanje koje se postavlja jeste da li kupac uopšte treba da zne koji čip – Intel AMD III Cyrix – čini srce njegovog novog računara. Magazin BYTE i NSTL (Nacionalna laboratorija za testirene softvera) su ispitali dve oblasti moguće diferencijacije brzinu i kompatibilnost. Obavljeni su i razgovori se brojnim ljudima koji zna ju kako rade popularni mikroprocesori

Testiranje smo obavili na mašini Tandon PC386 koja ima nekoliko pčmočnih kartica koje konste čelin osnovne procesora 33 MHz Intel 486DX AMD 25 MHz 386SXL i 40 MHz 386DXL i 25 MHz Cyrix 486SLC. Naši rezultati pokazuju da glavni procesor nije jedini faktor 25 MHz Cyrix je dao vid no bolja performansa od Am386SXL iste brzina (vi di sliku A) što ne treba de čudi, s obzirom da Cyrix ima procesorski keš na samom čipu. Medutim keš na procesoru nema mnogo uticaja na operacija koje nisu vezana za CPU kao što su operacije se baza ma podataka

U drugom testu smo proverili tvrdnju Cyrix-a da njihov procesol obavlje inbdžai množenja četin puta brže od konkurentskih. Sprovali smo lest sa serijom 16-bitnih množenja u pelji. Pokazalo se da je tvrdnja Cyrix-a tačna protok je poboljšan za oko 400% (vidi sliku B)

Naš lest je pokazao i neodrživost tvrdnji o povećanju brzina zasnovanih na testovima se malim kodom. Pri pisanju koda za testiranja dobili smo veoma različite rezultate – varirali su i do 50% – zavisno od položaje testiranog koda u našoj exa datoteci. To je zavisilo isključivo od toga da li se test kod izvršava u potpunosti unutar procesorskog keša ili ne. Keš se pokazao veoma bitnim i kod poređenja 40 MHz čipa AM386DXL sa 33 MHz čipom i486DX pošto AM386DXL ima 32 KB eksternog ka ša ne dodatnoj kartici

U cilju ostvarenja kompatibilnosti kompanija AMD koristi Intelov mikrokod Cyrix tvrdi da je sprovo pet potpunu implemantaciju 80x86 procesora koji je testiran ne signalnom novou kako bi se garantovala puna kompatibilnost se Intelovim zahtevima.

Mnoge kompenije kao što je proizvođač BIOS-a „Award" kažu da nikada nisu imali potrebu de promene ni jednu liniju kbas zbog nekompatibilnosti se non-Intel čipovima. Sve eventualne teškoće u identifikacji „Cyrix-ovog" čipa se rešavaju pomoću posebnog identifikacionog registra koji obazbeduje sem Cyrix Džefri Flink (Jeffrey Flink), inženjer Award-a kaže da razlikovanje čipova i potreba za pomoćnim čipovima koji procesori povezuju s ostat kom kompjutera predstavljaju izazev za projektante.

Ista tako ni nekoliko ICE (in-circuit emulator) proizvođača sa kojima smo razgovarali nisu naišli ni na kakve posebne probleme u pogledu kompatibilnosti se AMD i Cyrix procesorima – kompatibilnost je potpuna (nožica na nožicu). Ipali još uvek niko ne podržava specifične funkcije (npr upravljanja strujom i internl keš) koje nude Cyrix čipovi

Ovlašlih smo NSTL da obavi testiranje kompatibilnosti četiri Tandon procesorska modula. Umesto aplikacionog softvera testiranje smo zasnovali na usiušnim programima, kod kojih je veća verovatnoća da će zagazili u rede korišćene putanje procesora

Prvi problem se javia kod softvera koji pokušava da otkrije tip procesora na kojem se izvršava. Kod „Cyrix-ovog" čipa, tri programa su „prepoznala" različite procesora. „Nortonov" *SysInfo* je smatrao de se radi o 25 MHz Intelovom 486DX. *PCTools SI* je javio da se radi o 25 5 MHz Intelovom 486SX, *Control Room* je smatrao da je to 50 MHz 386. Ovo ukazuje na problem identifikacije novih procesora pomoću starog softvera i trebalo bi da poslužl kad upozorenje proizvođačima softvera da izbegavaju kod koji zavisi od brzine procesora

Još ozbiljnjih problem se javio kod dela paketa *PCTools* verzija 7 1 („Central Point Software"). tekozvanog *CPBackup* Naime pre no što napravi backup *CPBackup* obavlje proveru DMA (Dynamic Memory Acces) kontrolera. Ovaj test dovodi do zamrzavanja računara kod minimalno konfigurisanog sistema (tj kada nisu učtani drajveri uređaja) Po rečima Inženjera kompanije Central Point ovaj problem prouzrokuje vremenska petlja u softverskom kodu (U novoj verzji paketa problem je rešen tako što je uklonjana vremenska petlja koja zavisi od procesora.) Mada ovo strogo uzevši nije problem kompatibilnosti jer se ne radi o tome da li se ova lil ona naredba pravilno izvršava, on ipak ukazuje na teškoće koje se mogu javiti pri promeni konfiguracije procesora. Pokretanjem bilo kog upravljača memorije kao što je QEMM ovaj se problem u pot punosti otklanja

Poslednji uočeni problem ukazuje na to da pojedinačnu pažnju treba posvetiti kompletnoj interakciji sistema i svakog pojedinog procesora. Problem se pojavio kada smo pokrenuli *Softice*, program fi-

zvođača PC računala. Samo 25% je izjavilo da sare đuje isključivo sa Intelom. Neki su najavili made još ne isporučuju sistema sa čipovima koji nisu proizve đeni u Intel-u. Drugi opet kao na primer „Dell" name ravaju da i dalje posluju isključivo sa Intelom „Intelovi čipovi nude performanse koje nam trebaju i cene koje smo spremni da platimo" kaže jedan od direktora „Della"

KOLIKO ZNAČI IME?

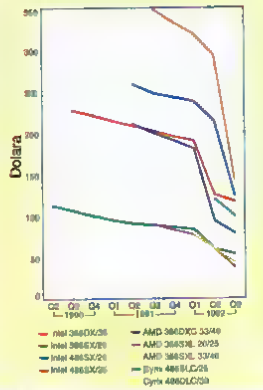
Proizvođači čipova se i ne trude da korisnicima omoguće direktna poređenja. U stvari, kao da njihovi stručnjaci za marketing namerno stvaraju zbrku. Na primer Cyrix je svoje čipove nazvao 486SLC i 486DLC. Međutim ovi čipovi nemaju matematički ko procesor imaju drukčiji raspored nožica i mnogo ma nji keš od Intelove 486-tice. Ipak, podržavaju set na redbi 486-tice dok imaju množenje obavljaju četiri puta brže od mikroprocesora klase 386.

Proizvođači i prodavci PC-je znaju u čemu je razlika, ali većina korisnika ne zna. Cyrix ovi mikroprocesori su „386 tice sa znatno unapređenim performansama" kaže predstavnik „Acera" Čvang, i nastavlja da 486SLX ima performanse za oko 20-30% sporije od Intelovog čipa 486SX što potvrđuju i testovi obavljeni u BYTE-ovoj laboratoriji.

Bil Berkman (Bill Berkman) direktor marketinga u kompaniji „Mylex" (proizvođač sistemskih ploča) tvrdi da Cyrix-ove cene nisu konkurentne cenama procesora 386 koji dolaze iz Intela ili AMD-a. Naime 40-MHz Cx486DLC je dvostruko skuplji od 40-MHz procesora AM386DXL (mada je u Cyrix-ovu cenu uključen i zaseban matematički koprocesor). Konsnik bi verovatno radije platio još 100$ i dobio istinsku 486 mašinu.

Ipak, mnoge kompanije – Tandy, Tandon, Com-

POREĐENJE CENA CPU



Slika 1. Ovaj grafikon prikazuje kako su proizvođači kompatibilnih čipova kao što je AMD uticali na CPU tržište. Otkako je AMD stupio na scenu u martu 1991, Intel je snizio cenu 33 MHz 386DX čipa sa $208 na $107. Sve Cyrix-ove cena uključuju i poseban matematički koprocesor

puAdd i U.K. Opus – konsle Cyrix procesore. „Za nas je Cyrix nova avantura" kaže Peterson iz Tandy-ja. Imajući na umu prijem kod potencijalnih kupaca „Tandy" se odlučio za Cx486SLC jer je odgovarao njihovim zahtevima kako svojim performansama tako cenom.

Predstavnici Intela su sveseni da se ljudi teško snalaze u moru raznoraznih mikroprocesora. Zato su uveli poseban sistem za merenje performansi verujući da će tako izbeći situaciju, što je Comp (*Intel Comparative Microprocessor Performance*) test brzina koji jednim indeksom izražava ukupne performan se mikroprocesora. Osnovna CPU platforma 25 MHz 486SX ima indeks 100. Test je namenjen brzom iz giranju pri svega Intelovih procesora, ali se može primeniti i na sve ostale kompatibilne procesore.

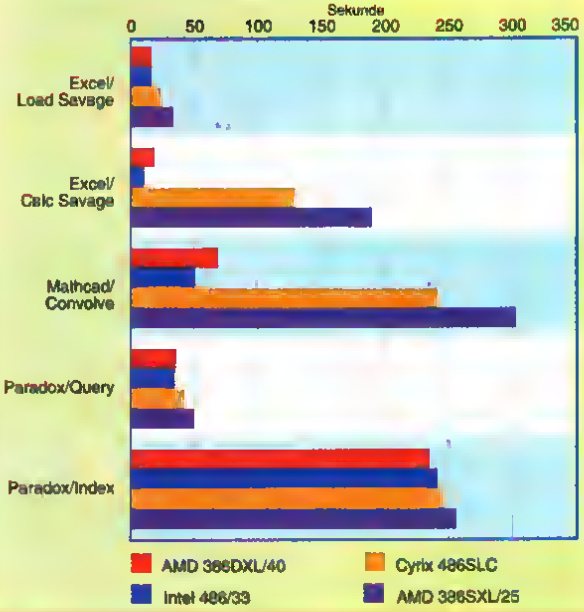
Konvencija po kojoj se imenuju proceson dodatni je izvor zabune. Proizvođač čije ime sadrži 486 (ili samo 4) ili nema jasne indikacije da se radi o Cyrix ovom čipu, odvešće kupca na krivi put. Na primer, Com puAdd ima čitav niz Intel i Cyrix mašina uključujući i 425SLC, 425SX, 433i i 433DLC. Pošečan kupac verovatno neće razlučiti da dve od ovih mašina imaju Cyrix-ove procesora.

Intel će i ubuduce imati problema s nazivima jer Amerčki zavod za patente ne prihvata kao zaštitni znak ime koje se sastoji od tri broja (npr. 586). Dakle ništa ne stoji na putu ostalim proizvođačima da svoje čipove lakođe nazovu 586 u namen da stvore konfuziju koja njima ide na ruku.

RAZMIŠLJANJA O DOGRADNJI

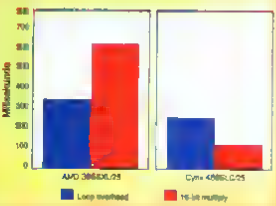
Pri izboru mikroprocesore treba voditi računa o mogucnosti dogradnje performansama i ceni. Pronalaženje prave kombinacija ovih elemenata usko je ve-

CPU PERFORMANSE



Slika A. Kao što smo i očekivali, Intel 486/33 je brži pri većini operacija. Brži radni takt AMD procesora deje mu prednost u drugim oblastima

POREĐENJE BRZINE 16-BITNOG MNOŽENJA



Slika B. Provera Cyrix-ove unapređene jedinice za 16-bitno inteldžer množenje. Mada se operacije sa „sirovim" brojevima obavljaju i više nego četverostruko brže, ubrzanje se jednim delom postiže zahvaljujući izvršavanju unutar keša jednog čipa. Kada to uzmemo u obzir, ubrzanje je nešto manje od četverostrukog koje navodi proizvođač.

mu „Nu Mega" (dibager na mašinskom nivou – low-level) na AM386DXL/40 procesorskom modulu. Prolaskom kroz testirani uzorak koji poziva BIOS funkciju „čekanja događaja" (Event Wait – INT 15h funkcije 83h) dolazi do prekoračenja internog steka i pada sistema. Kada je isti program pokrenut, pu nom brzinom iz samog *Softice-a*, problem se nije javio. Ovom prilikom nismo uspeli da izolujemo uzrok. Inženjeri firmi „Nu Mega" i „Tandon" bezuspešno su pokušali da isti problem prevaziđu na drugim sistemima, koristeći Am386DXL/40.

Da zaključimo, smatramo da su testirani procesori kompatibilni sa Intelovim standardom. Brzina i performanse zavise od ukupnog sistemskog okruženja, kolika i od samog procesora. Naravno, 40 MHz procesor je brži od 33 MHz, ali se razlika može ublažiti dodavanjem eksterne hardverske podrške. Na kraju, kupovina od proverenog proizvođača ipak ostaje najbolji način da se osigura potpuna kompatibilnost.

zano za oblik sisteme kome je namenjen. Mogućnosti dogradnje najvažnije je za desktop korisnike – mada se i na portabl sistemima već vidaju OverDrive utičnice tako da se na Intel 486SX sistem može installirati 486DX CPU.

Dogradnja omogućava da vaš stari sistem drži korak sa stalnim unapređenjem performansi mikroprocesora e da pri tom ne morate menjati celo kućište. Za nekoliko stotina dolara 16 MHz 486SX PC računar možete pretvoriti u 50 MHZ 486DX (vidi „Razlozi za dogradnju"). Međutim još uvek je mali broj korisnika koji su se odlučili za dogradnju, jer se poslednji tržišni ciklus još nije završio. Drugim rečima oni koji su kupili modele koji se mogu dograditi još nisu osetili veliku potrebu za unapređenim performansama.

Intel je napravio pionirski potez kada je uveo dogradnju u obliku jednog čipa: takozvani *OverDrive* procesor, koji udvostručuje unutrašnji radni takt procesora i povećava ukupne performanse za oko 50%. Pre pojave *OverDrive* čipa, dogradnje je ostvarivane zamenom matične ploče ili instalacijom dodatne kartice ili ploče. „Ljudi nisu hteli da daju 2000$ više za *AST Premium* (koji se mogao dograditi preko dodatne kartice) kada su za 1500$ mogli kupiti novu matičnu ploču. Međutim, sa čipom koji dvostruko ubrzava rad mikroprocesora, dogradnje je poslela jedan od ključnih elemenata pri kupovini.

Intelovi konkurenti tek sada ulaze na tržište dogradnji. Cyrix je nedavno predstavio verziju čipa 486DLC DRu² koji se konsb kao dogradnja za IBM i Compaq 386DX/20 sisteme (vidi sliku 2). U Cyrix-u kažu da će Dru2 raditi i se ostalim 386DX sistemima, mada iz mehaničkih razloga – dvostruko je viši od čipa Intel 486DX – nekim PC računarima neće odgovarati. Opcije čipa 486 DLC koje zahtevaju hardversku podršku, kao što je keš ili upravljanje strujom neće raditi optimalno ali će povećanje performansi ipak biti značajno. Najznačajniji element ubrzanje će biti Cyrix ova tehnologije udvostručenog radnog takta koja će omogućiti da čip podešen na 20 MHz ostven unutrašnju frekvenciju od 40 MHz.

BYTE je testirao prototipsku 33 MHz verziju DRu² na računaru IBM PS/2 Model 80. Sve u svemu, pokazao je 28% bolje performanse u odnosu na 16 MHz Intel 386DX. Video ubrzanje je variralo od 5% kod skrolovanje do čak 59% kod grafike. Ukupno unapređenje video performansi iznosi 35%. Kao što je i očekivano, testiranje diska na mašinskom nivou (low-level) je pokazalo najmanje poboljšanje – samo 10%. Mada su se delotekle učitavale i za 22% brže, opšti test koji je zatevao direktna učitavanja i zapisivanje je pokazao poboljšanje od samo 1%, budući da to zavisi od brzine drajva a ne od procesora.

Za sada, Cyrix nemarave da ovaj čip prodaje direktno velikim kupcima i ne planira maloprodaju. Procenjuju da će DRu² koštati između 200 i 300$.

U kompjuterskoj industriji vlada izvesni skepticizam u pogledu broja korisnika koji će se odlučiti za dogradnju. Džim Čepmen (Jim Chapman), zamenik direktora marketinga u Cyrix-u kaže da ako se samo 1% od 30 miliona 386DX mašine dograđi Cyrix će imati dovoljno veliko tržište. I AMD očekuje da tržište za dogradnju bude unosno – kompanije je javno izjavila da će podržati *OverDrive* utičnicu koju već imaju mnoge 486SX mašine.

BRZINE NIKAD DOSTA

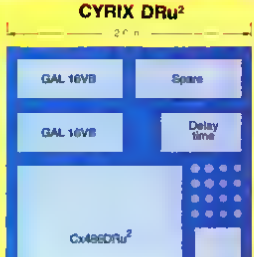
Razlike u performansama između 386 i 486 čipove takođe stvaraju zabunu. AMD se drži jedan korak ispred Intelovih linija 386SX i 386DX. Iako što uspeva da izbaci brže verzije svojih procesore 386SXL i 386DXL pre tntela.

Cyrix tvrdi da je ostavrio ubrzanje performansi pomoću aritmetičke jedinice koja obavlja intidžer množenje. Nanijel Julit (Daniel Oulette), zamenik predsednike kompanije „Microslate", koja proizvodi pen-računare, kaže da je testirao brzinu procesore Cyrix 486SLC i da se pokazao za 1 8 do 1.8 puta brzim od procesora Intel 386SX pri istoj trakvencij. Kod testiranja sa aplikacijom, razlike je bile još uočljivija. Pen-sistemi firme „Microslate" popularni su među korisnicime geografskih informacionih sistema. Julit kaže da je osvežavanje ekrane bilo 4-5 puta brže.

Do izlaske ovog broja Cyrix je verovatno već uveliko predstavio novu verziju popularnog čipa Cx486 Cx486S2/50. Za razliku od 486SLC i 486DLC 50 MHz Cx486S2/50 se udvostručenim radnim taktom odgovara 486 utičnici. Pošto nema matematičkog koprocesora, naziva se ekvivalentom čipa 486SX.

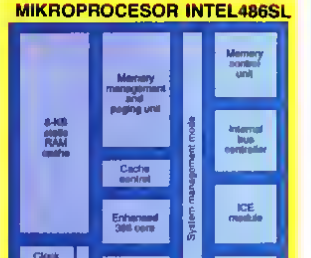
Cyrix tvrdi da će Cx486S2/50 imati unapređene

CYRIX DRu²



Slika 2: Cyrix-ov čip DRu² je direktna zamena (nožica na nožicu) za intel 386DX/20 SYTE-ov) testovi su pokazali de se performanse mogu unaprediti sa čak 28%. Glavni mikroprocesor je najveći kvadrat u donjem levom uglu. Ostale jedinice na pločici dimenzije 2x2.5 inča su pomoćni čipovi.

MIKROPROCESOR INTEL486SL



Slika 3: 486SL je Intelov 3,3-V mikroprocesor namenjen *notebook* sistemima. Osim brzine veće od procesora 386SL, nudi duže trajanje baterije. Pošto radi i na 5 V, bez teškoća se integriše u postojeće projekte 386SL mašina.

NEOBIČAN SLUČAJ FIRME NexGen

„NexGen Microsystems" (San Hoze Kalifornija) je po mnogo čemu zagonetna kompanija. Osnovan 1986. NexGen je počeo sa 386-kompatibilnim mikroprocesorom. Šest godina kasnije, NexGen još uvek nema gotov čip. 1990. se činilo da je na tragu nečeg značajnog kada je najavljen čip-set koji koristi CISC arhitekturu da bi postigao RISC performanse emulirajući 386/486 instrukcije. Kompjuteri sa ovim čip-setom bi radili dvostruko brže od 486 SPARC II. Mips sistema i približili se mašini IBM RISC System/6000. Nedavno je Nex Gen najavio P5-kompatibilan čip koji će se pojaviti samo mesec dana nakon što Intel izbaci svoj procesor nove generacije početkom godine.

Ovoga puta, posmatrači misle da gde ima dima, možda ima i vatre. Analitičari pretpostavljaju da bi NexGen mogao 486 jezgru koje navodno već ima, dodati paralelne intidžer jedinice proširiti keš promenivi radni takt i početi s prodajom „P5-kompatibilnog" procesora već sredinom 1993.

Majkl Selejter (Michael Slater) iz *Microprocesor Reports* kaže da NexGen nema P5 čip već samo brzu implementaciju seta instrukcija 486, koja se približava performansama čipa P5.

Iste mišljenje ima i Ken Loui (Ken Lowe) analitičar firme za istraživanje tržišta „Dataquest". „NexGen pokušava ono što je Cyrix uradio sa svojim 486-kompatibilnim procesorom. Upoređuju se sa P5 tehnologijom, jer žele da postignu P5 performanse. Dobar marketinški potez" Bilo kako bilo, ovo bi mogla biti poslednja šansa za NexGen da se dokaže.

NOVI ČIPOVI NA HORIZINTU

Tokom sledećih šest meseci možemo očekivati nove procesore glavnih takmaca na tržištu Intel-kompatibilnih mikroprocesora. Pošto proizvođači mikroprocesore rade u dosluhu sa proizvođačima sistema, te su oni u toku novih zbivanja mnogo pre zvaničnog predstavljanja, verovatno će se uskoro pojaviti i mašine sa novim procesorima.

Intel 486SX/33

486SX/33 je brža verzija postojeće finje procesora. Predstavljen je u septembru.

IBM 486SLC

IBM verzija procesore Intel 486SL. Predstavljen u oktobru.

Cyrix 33 MHz 486SLC

Brža verzija procesore SLC, sa čitavim nizom novih opcije za upravljanje strujom. Zamišljen kao direktna konkurencije Intelovom čipu SL. Predstavljen u oktobru.

Intel 486SL

486SL troši malo struje, a ima performanse procesora SX. Mnogi proizvođači *notebook* računara već su se odlučili za njega. Pojavio se u novembru.

Cyrix Dru²

To je 486DLC koji udvostručuje performanse 20 MHz 386DX sistema. Zamišljen kao dogradnja. Pojavio se u novembru.

Cyrix 486S2/50

486SX kompatibilan procesor sa manjim kašom nego kod Intelovog pandana. Cyrix tvrdi da je smanjenje keša nadoknađeno time što je on sada *write-back* tipa. Pojavio se u novembru.

Intel P5

Čip koji će zamenit 486 na vrhu Intelove proizvodne linije. Očekuje se u prvom kvartalu 1993.

AMD – čipovi klase 486

AMD obećave 25-, 33- i 50 MHz verzije svojih 486-kompatibilnih mikroprocesora, mada analitičari javljaju da će se verovatno pojaviti i čipovi sa udvostručenim unutrašnjim radnim taktom. Očekuju se do sredine 1993. zavisno od ishode sudskog spora u vezi sa korištenjem Intelovog mikrokoda.

Chips & Technologies PC/Chip se 386 jezgrom

C&T nije više na margini mikroprocesorskog tržišta. Novi čip će biti prvi rezultat ogromnih napora uloženih u proizvodnju 386-kompatibilnog procesora za „ručne" i male *notebook* kompjutere. Očekuje se tokom 1993.

Intel P24T

Intel P24T je procesor koji služi kao dogradnja za 486DX2 sisteme. Utičnice za ovaj čip već se ugrađuju na neke matične ploče. Utičnica je slična onoj za *OverDrive*, ali ima jedan rad nožice više. P24T će koristiti P5 tehnologiju za unapređenje performansi. Intel nije želeo da precizira šta podrazumeva pod P5 tehnologijom, ali je moguće da će P24T kao P5 imati poboljšan FPU i paralelne intidžer pejpove.

NexGen – ikon čipa P5

Radi se o čipu tipa P5 koji NexGen namerave da izbaci nakon što se pojavi Intelov P5. Poznavajući NexGen, možda ćemo čekati i do 1994.

REZERVNI IGRAČI OSVAJAJU NOVE SEGMENTE TRŽIŠTA

Veliki broj proizvođača mikroprocesora je najavio planove za proizvodnju Intel-kompatibilnih procesora za usko specijalizovane aplikacije a među njima su i „NEC Technologies" „Chips & Technologies" „International Mate Systems", „Vadem" i „NexGen"

Najstariji igrač na polju procesore kompatibilnih se Intelom je bio NEC sa svojom V-linijom procesora koja se pojavila 1984. Originalni mikroprocesori su projektovani u Intelu a NEC je dobio licencu. Kasnije je ova linija zamenjena procesorima projektovanim u samom NEC u. Danas NEC prodaje 30 verzija V procesore uključujući i nisko-voltne VHL 33 MHz procesore. Oni se nalaze u raznim mašinama od Olivettijeve *Quaderno SubNotebook* do Compaqovog IDA drajv kontrolera. Najnoviji je V55 procesor – kompatibilan se procesorom 80186 na 12.5MHz. Originalni mikrokod korišten za 8086 i 80186 je ugrađen u hardver tako da se većina instrukcija izvršava u jednom ciklusu.

„International Mate Systems" ima ambicioznija planove. IMS tvrdi da će novi 100 MHz Risc mikroprocesor, koji će početi da se proizvodi sredinom ove godine. Biće u stanju da emulira Intel 486 ili Motorola 68040 istom brzinom ali po znatno nižoj ceni. IMS ja navi čip namenio pen računarima koji zahtevaju visoke performanse za funkcije kao što su obrada i tumačenje rukopisa ali će biti pogodan i za „kameleonski kompjuter" koji će izvršavati programa pisana i za PC i za Macintosh.

U aprilu prošlog godine mali proizvođač čipsetova „Vadem" je predstavio mali PC zasnovan na jednom jedinom čipu, takozvani „palmtop" (računar koji stane na dlan). Proizvođači kompjutera mogu koristiti *VG-230 Sub-Notebook Engine* i vrlo brzo izbaciti na tržište jeftine „palmtop" računare. „Vademov" čip je projektovan tako da inkorporira 16 MHz NEC V30HL mikroprocesor u uređaj se jednim jedinim čipom, koji sabrži LCD kontroler, internu skener tastature i ugrađenu jedinicu za upravljanje strujom. PCMCIA 2.0 modul pruža periferni podršku. Poslednja verzija podržava i digitalno pero, tako da odgovara pen-računarima. Sekundarni prostor displej memorije prima trag elektronskog pera, tako da sistem obrađuje poteze i prepoznaje rukopis, na ulučći pri tom na originalnu sliku na ekranu.

C&T se takođe okreće tržištu „ručnih" kompjutera. U avgustu je kompanije najavila da će se usredsrediti na integrisane procesore i logičke čipove koji će proizvođačima PC računare pružiti rešenja u obliku jednog čipa za „ručne" i *notebook* kompjutere. C&T već ima PC/Chip integrisani procesor koji daje PC ossbine 8086 jezgru. Sada će razviti 386 jezgro za PC/Chip i VGA kontrolar koji će zameniti CGA displej drajver niske rezolucije. Gordon Kembel (Campbell) predsednik kompanije C&T tvrdi da će se veoma brzo pojaviti prvi ozbiljni sistemi sa baterijom koja će omogućavati 50 sati rada.

performanse zahvljujući *write-back* kešu koji daje veći protok (throughput) se manje keša. Cx486S2/50 će konstih 2KB keša što je ekvivalentno Intelovom *write-through* kešu od 8KB. *Write-through* keš procesor koristi samo kod *read* operacija dok *write* odlazi direktno u mnogo sporiju glavnu memoriju. Kod *write-back* tehnike procesor zapisuje promene u keš memoriju sve dok je ne napuni. Tada se ceo memorijski blok kopira u glavnu memoriju brzom *blokiranom* operacijom. Ne ovaj način se postiže znatno ubrzanje. Cyrix tvrdi da će performanse prosečne aplikacije biti jednake onim koje daje 50 MHz procesor Intel 486DX2 dok će performanse grafike biti unapređene za oko 10%.

Cyrix-ov čip će moći da pristupa memoriji u jednom ciklusu a ne u dva. Cena procesora Cx486S2/50 će biti oko 250$ za proizvođače sistema.

Odavno prisutno u minikompjuterskim i *mainframe* okruženjima, profiliranje instrukcija se javlja i kod mikroprocesora. Profiliranje instrukcija je analiza relativne frekvencije instrukcije koje konsle određene aplikacije. Sa ovim informacijama inženjeri mogu podesiti arhitekturu tako da je za češće korišćena instrukcija potrebno manje ciklusa. Čak i po cenu da se koristi više ciklusa za izvršavanje redih instrukcije čime se znatno unapređuju performanse.

Intel uveliko koristi profiliranje instrukcije pri projektovanju novih procesora. I tako stvara ogromnu bazu podataka koje sadrži pulanje instrukcije različitih komercijalnih aplikacija. Rezultat Intelovog profiliranja je i učena razlika u upotrebi instrukcija kod aplikacija pisanih jezikom C++ i onih pisanih drugim jezicima što opet ukazuje na blisku vezu između kompajlera i procesora. Ako C++ prevladaja na polju komercijalnih alatki za razvoj softvera Intel i ostali će se možda odlučiti da svoju arhitekturu podese prama frekvenciji instrukcija iz ovog jezika.

Razlozi za degradnju

- Boja performanse
- Ušteda novca u odnosu na kupovinu novog sistema
- Kraće vreme potrebnoi za zamenu
- Lakša odluka kod kupovine
- Lakše održavanje koreka e razvojem tehnologija

PERFORMANSE SA BATERIJAMA

Performanse dobijaju sasvim drugu dimenziju u areni prenosivih računara gde su štednje struja i visok stepen integrisanosti ključni faktori. Intel je planirao predstavljanje porodice 486SL koja troši malo struja. Prva dva mikroprocesora će raditi na 25 odnosno 33 MHz i oba imaju ugrađen matematički koprocesor. Intel kaže da je 486SL fizički manji i ima veći stepen integrisanosti nego 386SL što će omogućiti proizvođačima da smanje dimenzije *notebook* matičnih ploča za oko 60%. 25 MHz čip već se proizvodi, dok se 33 MHz verzija planira za prvi kvartal 1993.

486SL (vidi sliku 3) predsatvlja značajan *napredak* u odnosu na 386SL. Intel tvrdi da se performanse kreću od 11 MIPS-a za 25 MHz 486SL do 1.4 MIPS-a za 33 MHz verziju (radi poređenje najbrži 386SL (25 MHz) ima brzinu od 5.2 MIPS)

Jednako značajno je i svojstvo „fleksibilna voltaže" koje poseduje 486SL. Iako je 486SL 3.3 voltni statični uređaj može raditi i na 5 V. Zahvaljujući tome – kao i kompatibilnosti čipa 486SL sa SMM-om (modul za upravljanje sistemom) procesora 386SL i I/O čipom – već krajem godine možemo očekivati *laptop* računare sa ovim čipom. Dok će neki proizvođači jednostavno prilagoditi 486SL postojećim 5-voltnim matičnim pločama, naprednije mašine će raditi na 3.3V. Kada se 486SL prilagodi ostalim 3.3 voltnim delovima, potrošnja struje se smanjuje za 50% u odnosu na 5-voltni 386SL sistem. Intel kaže da bi ovo trebale da produži trajanje baterije za 1-4 sata.

Kao i 386SL, novi procesor 486SL ima PI (Peripheral Interface) sabirnicu koja je po svojoj funkciji skvivalentna lokalnoj sabirnici kod desktop kompjutera. PI sabirnica omogućava grafičkom kontroleru ili fleš-kartici da zaobiđu I/O sabirnicu i direktno se vežu za CPU. PI sabirnica radi punom brzinom mikroprocesora dok I/O sabirnica ima relativno sporu propusnu moć od 8MHz.

FAKTOR CENE

Konkurencija je smanjila cene čipova, kao i cene sistema. 25 MHz 486DX je prvobitno koštao 950$ i znatno je dopprinosio ukupnoj ceni sistema, sada košta 328$. Po rečima Steva Vorena (Steva Warren) zamenika predsednika kompanije „Altima Systems" koja proizvodi *laptop* kompjutere. CPU sada mnogo manje utiče na cenu sistema. „Altima" prodaja AMD 386SX sistem i Intel 386SL sistem. AMD sistem je 10% jeftiniji od sistema sa Intelovim mikroprocesorom. Ali zato ovaj drugi portabi daje bolja performanse zahvaljujući *eksternom* kešu.

Kad je u pitanju cena Intel ima značajnu prednost u odnosu na svoje konkurente: kooperativnu reklamu. Nelma. Intel plaća proizvođačima sistema da podrže kampanju pod nazivom „Unutra je Intel" kada reklamiraju svoj proizvod. Ovakav aranžman je posebno privlačan za kompanije koje vrše prodaju putem narudžbenice jer njihov prihod potpuno zavisi od reklame. Dva britanska proizvođača, „Elonex" i „Viglen" kažu da Intel snosi 5% ukupnih troškova za reklame u novinama i časopisima – jedini uslov je da reklamni oglas sadži i logo „Unutra je Intel". Jedan američki proizvođač pominja sličan ugovor ali dobija od Intela samo 3%. Za mnoge proizvođače ova pogodba uvelike nadoknađuje razliku u ceni između skupljih Intelovih i konkurentskih čipova.

Mnogi konsnici i proizvođači imaju više poverenja u Intel. Mark Vena, direktor proizvodnje kompanije „Epson America" kaže „Ako bi dva procesora – AMD i Intel – imali istu cenu. Ipak bismo uzeli Intelov zbog zajedničkih resursa i dosadašnje saradnje."

Isto stanovište zastupa i proizvođač softvera „WordPerfect" Piter Mogan (Peter Maughan) je zadužen za održavanje mreže i nabavku radnih stanica za potrebe svoje kompanije. Testira svaku mašinu koju „WordPerfect" kupi. On kaže „Najmanje problema su nam stvarale mašina sa Intelovim procesorima."

PRILAGODENE FUNKCIJE

Sistemi postaju sve raznolikiji (vidi sliku 4). Oblasti koje su se nekada smatrale marginalnim delovima tržišta sada su u centru pažnje. Na primer, udeo koji na tržištu zauzimaju desktop računari je smanjen otkako su *notebook* računari postali dovoljno moćna e time i sasvim prihvatljiva alternativa. Predvidanja koja daje „InfoCrop" firma za istraživanje tržišta, pokazuju da će prodaja *notebook* računara porasti za 22.7% u toku sledeće dve godine dok su u 1992. godini zauzeli samo 14.5% ukupnog PC tržišta.

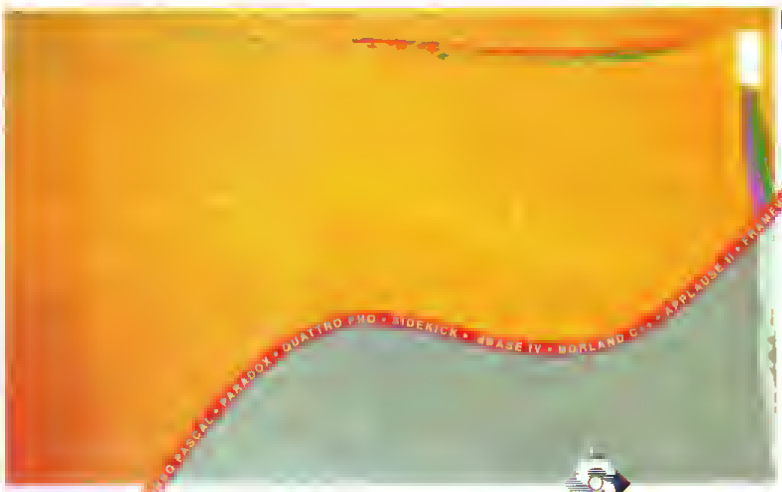
Intel je prošlo svoju ponudu mikroprocesora da bi konkurisao ostalim proizvođačima na tržištu specijalnih kompjutera. O tome svedoči 25 verzija procesora 486 koji treba da se pojave tokom 1993. godine. Veliki je broj procesora koje će proizvođači sistema podržati. Procesori će biti sve brži i trošiti sve manje struje. AMD i Cyrix će imati čipove potpuno kompatibilne sa 486-ticom, a čak i Intelovoj velikoj nadi, P5, prete NexGen i ostali konkurenti. Po svemu sudeći, neizbežno će doći do smanjenje Intelovog udela na tržištu, a time i do smanjenja profita.

Nastavlja se na stranu 34

Slika 4. CPU se sve više projektuje tako da odgovara specifičnim potrebama svake od kategorije sistema. Ručni kompjuteri zahtevaju visok stepen integrisanosti i mali utrošak struje. Portabl sistemi traže i performanse i štednju struje. Desktop PC traži visoka performanse pr prihvatljivoj ceni. Konačno, radnim stanicema i serverima treba čista kompjuteraska snaga

MISOFT OVLAŠĆENI DISTRIBUTER KORPORACIJE MICROSOFT
11000 BEOGRAD, SKADARSKA 45 TEL 011/343-043

Design Labyrint • Priprema B&Z



HIGH YELLOW JULES OLITSKI

SOFTLAND
OVLAŠĆENI SUBDISTRIBUTER KORPORACIJE BORLAND
11000 BEOGRAD • SKADARSKA 45 • Tel 343-043

Sistem Integrator

- *RISC tehnologija od grafičkih radnih stanica do superkompjutera paralelne arhitekture*
- *UNIX/IRIX, LAN/WAN TCP/IP, Relacione baze podataka, CASE alati*
- *Kompletan inženjering poslovnih informacionih sistema*
- *Aplikativni softver: proizvodnja, finansije, bankarsko poslovanje*
- *3D – računarska grafika CAD/CAM arhitektura geografski informacioni sistemi, vizuelno procesiranje, dizajn, animacija*

SiGraf

SiliconGraphics
Distributer

Si Graf • Računarska grafika i mreže • Niš • Bul. Vojvoda Mišića 80 • Tel. (018) 342–132 • Fax (018) 341–466

MOJ VIRTUELNI SVET

Da je dr Milan Božić atraktivan sagovornik, čovak koji se na iibi, ume i ima šta da kaže - bilo je jasno već onog trenutka kada je, pod višobranom ciničnog alogana „baviću se politikom da se ona na bi bavila sa mnom" - uzburakao domaću scanu i izvan ezoteričnih krugova univerziteta. Matematičar, arudita, ljubitalj fiiozofija i naučnik u najboljoj tradiciji ariatotelovaka škola mišljanja - dr Milan Božić a raziogom izazive pažnju kako pokionika, aledbanika i saradnika, tako i ljutih protivnika. Ravnodušnih nama.

Razgavar vadila: Vesna Ćosić

Kada je 1950. legendarni britanski matematičar i logičar Alen Tjuring zamislio čuveni test u kojem čovek sedi sam u sobi povezan teleprinterom sa drugim čovekom ili mašinom (pitanje je sad), dajući sve od sebe da otkrije sa kim (ili čim) je u vezi – kontroverze mogu li kompjuteri da misle stavljena je na dnevni red

– Kontroverza poverana se pitanjem „Mogu li kompjuteri da misle?" koja je kako Vi kažete, iskrsle u vezi za Tjuringovim zamišljenim testom iz 1950 mnogo je starija držim nego ovaj vek!

Ovo što sam rekao može delovati paradoksalno ali kada malo bolje razmislimo videćemo da je to pitanje u stvari preformulisano pitanje koje glasi „Mogu li mašina da misli?" ili još „filozofskije" „Mogu li nežive stvari da misle?" Tako gledano ona je staro koliko i ljudski rod ili bar koliko filozofija sama, ako je shvatimo kao pokušaj samorazumevanja.

Moja filozofska pa prema tome i epistemološka pozicija je takva da na ovakva pitanja uvek odgovaram onako kako zamišljam da bi odgovarao autentični pozitivista. Držim dakle da je ovde po sredi problem značenja glagole „misliti" ili imenice „misao" ako je ona starija. Problem je u tome što u svima nama postoji jaka intuicija, a neka starija filozofske škole su je zdušno podržavale i obrazlagale da je mišljenje osobina ljudskog roda i eventualno, nekih životinja, živih bića jednostavnije rečeno. Ako je tako onda, naravno kompjuteri ne mogu da misle

Međutim prethodni odgovor je prilično sirićan, jer mi veoma dobro znamo de onaj koji mišljenje pripisuje samo živim bićima, prećutno, pretpostavlja da su živa bića nešto „veoma posebno" da su obdarena duhovnošću dušom ili nečim sličnim te da zato i mogu da misle Ovu pretpostavku to bar danas dobro znamo savremena nauka veoma uspešno otklanja. Živa bića se u osnovi sastoje iz nukleotida i proteina – analogija sa softverom (nukleotidi) i hardverom (proteini) je prosto frapantna, ali je treba ostaviti u domenu „kompjuterske poetike" da ne bi izazvale nove rasprave – te su preme tome sasvim „obične" ali naravno, veoma komplikovane „prirodna pojava" – kaogod i kvazari koji bar mena znatno više impresioniraju Pri tome naravno otklanjam gotovo kao vic, tezu da se živa bića razlikuju od kompjutera po tome što su nastale kroz evoluciju koja je trajaja vile miliona ili milijardi godina, jer to nije ono o čemu mi ovde govorimo

Ako se naš hipotetični sagovornik sa kojim opet hipotetično polemišem složi sa prethodnim onda nam preostaje da razjasnimo značenje reči „mišljenje" Jedino mu još tu ostaje nada da capon tezu da kompjuteri mogu da misle E sada je on u neprilici s ne jel Jer on raspolaže samo veoma uopštenim definicijama tipa „mišljenje je sposobnost snalaženja u novim situacijama" i slično Tu sam je „na svome", jer uvek mogu tvrditi da i kompjuteri to mogu Ukoliko pak, pribegne „poslednjem sredstvu" i počne da nabraja naše posebne zakljućke do kojih se stiže „pomoću mišljenja" tek ja onda aktiviroo svoje pravo oružje! Sasvim je izvesno de kompjuteri mogu postići mnoge toga sto i mišljenje izvesno je takođe da neke stvari ne mogu postići na primer je ću u ulični guzvi lako prepoznati svog prijatelja lii rođaka e to kompjuteri ne mogu

Međutim i tu je kraj moje priče nema principijelnih prepreka da to jednom na bismo mogli Ukratko držim da kompjuteri mogu da misle ukoliko je svet koji nas okružuje uopšte kauzalan, ali bar za sada to čine prilično rudimentarno!

Sajmor Pejpert sa MIT-a misli da se veštačka inteligencija može uporediti sa veštačkim letom. hipotetički skeptik bi po njemu rekao – Vi matematičari imate posla sa zamišljenim fluidima – stvarna atmosfera je neuporedivo složenija! Međutim, okreni – obrni isti principi odnose se na stvarne i zamišljene fluide i primenjivi su bez obzira da li fluid struji preko perja ili aluminijumskog krila.

– Poređenje je sasvim dobro odlično ćaki Dopustimo hipotetičnom skeptiku da je u pravu Međutim on ovom tvrdnjom ne dokazuje ništa! Naime avioni ipak lete i to je valjda bio glavni cilj braće Rajt a i svih poslenika avijacije kasnije Oni nisu pravili ptiću oni su pravili – i uspeli da naprave – letilicu!

Na isti način oni koji prave računare – pri čemu ovaj izraz upotrebljavam u najširem smislu odnosi se i na hardver i na softver – na pretenduju da napreve mozak – ni pileći a nekmoli ljudski, bez obzira što su se u vreme mog detinjstva kompjuteri nazivali „elektronskim mozgovima" – već spravu koja će obavljati izvesne jasno definisane poslove koje može da obavlja i ljudski mozak

Tako kao što avion nije ptica već „ume" da radi nešto što ume i ptica – to jest da leti – ali ne ume da

Živa bića se, u osnovi, sastoje iz nukleotida i proteina – analogija sa softvarom (nukleotidi) i hardverom (proteini) je prosto frapantna.

nosi jaja, tako kompjuter ume nešto što ume i čovek – recimo da obavlja osnovne aritmetičke operacije što svakako nije spomo – ali ne ume da se recimo, poli tički angažuje

Takođe kada se e ovakvim stvarima raspravlja valja stalno imati na umu da pridev „veštački" koji počiće od na baš najsređnijeg ali jedino mogućeg prevoda engleskog prideva „artificial" u engleskom jeziku ima negativan vrednosni prizdnak. Taj vrednosni prizdnak mada postoji i u engleskom jeziku nije u njemu toliko jak. „Artificial" označave više nešto što je „napravljeno" e ne nešto što je „veštačko" Veštačko mleko veštačke grudi ili veštački zubi nisu kod nas Srba neke osobito cenjena stver Moje insistiranje na ovom detalju može delovati pomale neobično ali imam utisak da na malo neoporuzuma i psihološke odbojnosti prema „veštačkoj inteligenciji" leži i u nerazumevanju onoga što joj je stavljeno u zadatak. I u zameni teza, prema kojoj joj se pripisuje nešto No ona svakako nije

Dobro može li hrpa silicijuma i programa da dosegne mozak, jer ako jednom rast znanja čak i sazna sve zakona i principa koji kontrolišu svest-duh-dušu može li se ona duplicirati? Uopšte zapitćemo li se u paučinu filozofske razudjenosti roneći kontroverzeme veštačke inteligencije?

– Najiskrenije i svakako najtačnije, bi bilo da odgovorim „Ne znam!"

Ipak to me ne spređava da malo spekulišem.

Pre svega, čini mi se da je prarano postavljati to pitanje matematičarima inžinjerima kao i ma kome ko se bavi računarstvom Kažem „prerano", jer nama još uvek nije formulisan zadatak. Šta mi to treba da uradimo ili napravimo da biste nam vi rekli „E ove hrpa silicijuma i programe je dosegla mozak"?

Precizuje, to znači da biologija i medicina – o psihologiji ne bih ni govorio jer se ona veoma teško može svrstati u nauka to je više neka vrsta esejistika koja pretenduje na egzaktnost – još uvek nisu popunile nedostajuće pločice mozaika I mi nemamo tačno definisan I formalno opisan objekat koji sa naziva „mozak". Ne tvrdim da je to upšte moguće ali bi se do rezultata najverovatnije moglo doći simbiozom, sintezom računarstva sa biologijom i medicinom.

Tu leži odgovor i na drugi deo Vašeg pitanja, jer mi se čini da će „filozofska razuđenost" biti neizbežna posledica radikalne promene pojmovnog aparata koji će se tokom sledećih nekoliko decenija razvijati kroz pomenutu simbiozu

Mi, recimo na sasvim legitiman način danas razumevamo elementarne čestice u fizici kroz aparat kvantne mehanike ali je taj aparat svakako nedostupan filozofskom mišljenju poteivljenom na klasičan način

Ako je Tjuring bio u pravu i savršena simulacija razmišljanje jeste razmišljanje – šta reći za mišljenje filozofe Denijela Denela koji je ovakvo stanoviste nazvao „visokom crkvom kompjutacionalizma" u njene prisate strpao ličnosti kao što su Herbert Sajmon i Marvin Minski a legendarni MIT etiketirao kao svejevrsni Vatikan?

– MIT je nema sumnje najugledniji svetski centar za „poslove ove vrste" U tom smislu on jeste „svojevrsni Vatikan" Takođe sasvim je razumljivo što u savremenom svetu koji sva više postaje ako već i nije poeted, Maklusanovo „globalno elektronsko selo" reklama, uspeh i publicitet igraju ogromnu ulogu te se kolegama se MIT e na može zameriti što svoje rezultate i uspehe kao i svoje pogiede na problematiku, glasno i agresivno zastupaju

Čini mi se međutim da je Denetovo mišljenje što je izgleda „bolest" svih filozofa koji su vesptavani da misle globalne znatno pretenciozuije. Ono sadrži tezu da je MIT ili krugovi bliski njemu formirao nekakvu dogmu koju brani na način kako te čine crkve.

Ova teza naprosto nije tačna.

Prvo prilično je sporno da uopšte postoji nekakva dogma koja bi se svodila na tezu „savršena simulacija razmišljanje = razmišljanje" već bih pre rekao da je ono što se MIT u s i celoj savremenoj nauci može pripisati u stvari, jedan gnostički prisiup. Naprusio postoji u čiavo, svetskoj nauci jedno jako ubeđenje da se svet može spoznati Ovo ubeđenje je epistemološkog karaktere i svakako ne spada u dogme Kanačno u dosadašnjoj istoriji nauke nije se dogodilo ništa što bi nas razuverilo u ovo predstavu Naprotiv!

Drugo aktivnosti koje eventualno proizlaze iz ove pretpostavljene dogme, sigumo nisu takve da idu na sprečavanja ili osujećivanja napora druge vrste, mada – neka mi bude dopušteno da budem pomale sarkastičan – nije mi jasno koji bi to „napori druge vrste" uopšte mogli da postoje?

Zajedljive knjige poput one Hjuberta Drajfusa „Šta kompjuteri ne mogu da rade" – izazivaju talase polemika ali i zanimljivih istraživanja.

– Svakakol Uvek je za jednu nauku važno da ima puno zajedljivih kritičara. Istina, oni *mogu* zadati grdne muke pojedinim naučnicima, ako se ne njih lično okome ali za nauku kao celinu, oni su od neprocenjivog značaja Značaj takvih ljudi u savremenom svetu što je pomalo paradoksalno raste Naime, savremenu nauku karakteriše praktično nestajanje naučne kritike Ova pojava je posledica razuđenosti nauke, pojava esnefskog i ekspertskog žargona i, konačno ali ne i najmanje važno dramatične masovnosti savremene nauka Naprosto javnost je i nerazumljivo i nezanimljivo bilo kakvo javno vrednovanja neučnih rezultata. Ono se obavlja unutar profesije poneksad po ispravnim a ponekad po krajnje sumnjivim kriterijumi-

Držim da kompjuteri mogu da misle, ukoliko je svet koji nas okružuje uopšte kauzalan, ali, bar za sada, to čine prilično rudimentarno!

Fotografija: Studio Tekić & Nenad Petrović

Dr Milan Božić: "Ovaj naš razgovor obaćnom čoveku deluja kao razgovor ludaka. Voleo bih da nam se u 1993. dogodi, ne naređ, deste nam ga je bilo, nego vreme u kome ovaj intervju neće biti razgovor ludaka."

ma, ali ne dopira do javnosti i tako nauka ostaje bez čuvene povratne sprege koje ju je podstrekla na razvoj sve do prvih decenija ovog veka.

Ljudi kao što je Drajfus upravo obavljaju ovo značajne posredovanje između javnosti i nauke. Oni to rade na prilično uprošćen popularizatorski način i ne mogu izbeći kritiku da ovim načinom iskrivljuju stvarnu sliku, ali se se razlogom možemo zapitati: „A kako bi inače mogli da stignu do šire publike?"

Kompjuterski program je ipak nešto više od talasa elektrona. Ako ga stavite u drugi kompjuter, gde je struktura silicijuma i metala potpuno različita, pa je i način kretanja elektrona različit – program je isti čini isto. Da li to znači da smo i mi samo procesori neuronskih simbola? Jer i pojedinačni molekuli enzima,

Pridev „veštački" u arpakom jeziku ima negativan vrednosni predznak. Veštačko mleko, veštačke grudi ili veštački zubi nisu kod nas Srba neka osobito cenjena stvar.

ljudi i DNA od kojih je sačinjena ćelija prilično su jednostavni, ponašaju se po poznatim zakonima fizike i hemije i ni jedan od njih se ne može izdvojiti i nazvati živim...

– Svakako da se mi ne možemo nazvati „procesorima neuronskih simbola" (bez obzira što je izraz nejasan, mislim da otprilike shvatam na šta ste mislili) zbog puke okolnosti da isti program, kada promeni „stanište" tj. kompjuter radi opet isto. Međutim analogija između nas i kompjutera se ne može izbeći one nam se šta više sve više nameće kako savremena biologija, molekularna pre svega, napreduje.

Pre svega slike živih organizama, koju nam daje savremena molekularna biologija, neodoljivo podseća na kompjutere.

Centralna dogma molekularne biologije tvrdi: sva teorijska i eksperimentalna istraživanja idu u prilog ove teze, da genetsku informaciju nose samo nukleinske kiseline a ne proteini koji služe kao nekakav hardver. Nukleotidi, osnovne jedinice od kojih su sastavljene nukleinske kiseline naposredno koreliišu sa tokenima nekog programskog jezika.

Osim toga svi eksperimenti u koje je poslednjih decenija investirano i mnogo pameti i mnogo novca, pokazuju da se replikiranjem nukleinskih kiselina, bar na nivou jednostavnih organizama, mogu replicirati i ti organizmi što je krunski dokaz o hemijskoj ili – ako smatrate da je to zasebna disciplina – bio-hemijskoj osnovi života koje se može opisati bez ostatka.

Uspesi koje je postigla molekularna biologija su u toj meri učvrstili naše uboćenje da između kompjutera koji izvršava neki program i živih bića nema nikakve principijalne razlike da je vrlo teško pronaći naučnika koji bi, danas, tvrdio suprotno.

Što se odgovora na poslednji deo pitanja tiče, on je u svetlosti ovakva metodologija jednostavan – ni procesor ni memorija ni sredstva za masovno skladištenje podataka, ni sami programi ni periferni uređaji nisu, uzeti svaki za sebe radeći kompjuter. Njega čine sve ova komponente zajedno i to samo ako su funkcionalno povezane.

Nije tajna, digitalizacije je učinila komuniciranje između ljudi i mašina, između mašina, pa i mašine sa ljudima lakim kao što je to ono među ljudima. Da li je dan kada da i ukusi mirisi i što da ne osećanja, moći da se pohranjuju obrađuju i prenose – već tu negde iza ugla? Ili je bard SF-a Isak Asimov bio u pravu tvrdeći da „ništa ne može da zameni iskustvo iz prve ruke"?

– I ovde pošten, iskren i, pre svega tačan odgovor mora da glasi „Ne znam!" Naime okolnost da život, terijski i principijelno za nas više nije tajna ne mora da znači da ćemo mi ikada, moći se njim da „radimo šta god hoćemo". Tako, na primer, možda mi nikada nećemo moći da napravimo „veštačkog čoveka" tj. živo biće sintetizovano u laboratoriji koje do ne molekul. I što je mnogo važnije, do ne njihov razporedi odgovara nekom drugom čoveku.

Jednom rečju posedovanja informacija o nečemu ne povlači automatski posedovanje tehnologije da se to „nešto" kontroliše ili replicira.

Tako, na primer, mi danas gotovo savršeno poznajemo zakonitosti nuklearne fuzije i znamo da je ona potencijalni izvor ogromnih energetskih resursa, koji bi ne duga staze, a možda i zauvek rešili sve naše energetske probleme. Međutim nikto do sada ni-

Uvek je za jednu nauku važno da ima puno zajedljivih kritičara.

je napravio fuzioni reaktor. Šta više, nisam uopšte siguran da će on ikada biti napravljen. A evo i zašto. Pod fuzionim reaktorom se podrazumeva uređaj koji radi u linearnom dakle, stabilnom režimu a ne u impulsnom – takva „uređaja" već posedujemo, to su nuklearne bombe, ali ne verujem da da se ma čiji energetski problemi rešili ako mu spustimo jedan takav „uređaj" na glavu – i prolivanje više energije nego što se u njih ulaža. Sasvim je moguće da da se jednog dana ispostavi izračunati i dokazati da je snalnje potrebna za održavanja stabilnog polja od stotinak miliona stapeni veća od energija koju nam takvo polje fuzijom ćestica koje su u njemu vreće natrag, i naš san o fuzionom reaktoru da biti raspršen.

Slično tome možemo zamisliti hipotetičnu budućnost u kojoj će postojati mogućnost da se prikupe sa-

Šta mi to treba da uradimo ili napravimo da biste nam vi rekli „E, ova hrpa alicijuma i programa je dosegla mozak"?

solutno sve informacije koje, na primer ja, u sebi nosim, ali se ipak neće moći napraviti uređaj koji će po liku prepoznavati moj komšiju u me kom realnom vremenu, dok ja to činim, što ta rekli, „u trenutku".

Lično ne mislim da je takva budućnost stvarna i imam utisak možda preterano optimistički da je manje-više sve što se može saznati i ostvariti.

Naša mašta teži da se vezuje se ograničenja iz prošlosti vidi je sutrašnjice često je obazrivo konzervativna i podseća na istraživanja koje je Mercedes Benc sproveo 1900 godine u kome se predvidja da svetska potražnja za automobilima nikada neće preći cifru od million pre svega zato što neće biti dovoljno šofera.

– Prema takvim pitanjima imam dvojak odnos. Sa jedne strane nema nikakve sumnje da je naša stvarnost nadmašila i najluđe snove naših predaka. Tako gledano, maštanja a la Žil Vern treba ne samo podržavih nego ih vaja i podsticati. Lično sam uboćen da Žil Vern nije samo „predvideo" neka od čuda savremene tehnike već je i sam uticao ne njihovo stvaranje – neki od dečaka koji je čitao njegove knjige sigurno je kasnije postao naučnik ili inžinjer, pa je, voden se-

Mislim, dakle, da će civilizacije biti najsrećnija i najuspešnija ako Zemljinom kuglom, ravnopravno, budu hodili i veliki sanjeri ali i skeptici.

ćanjima ne literaturu svog detinjstva, bar nešto od toga pokušao da ostvari i, bogami, ostvanoi.

Sa druge strane, neodmerena maštanja mogu vući istraživanja pa i čitave kultura na stranputicu. Pri tome ne mislim na budalaštine Denikenovskog tipa, jer to je ipak literatura za razbibrigu, već na promašaje ozbiljnih veličina kao što je, na primer, bio Nikola Tesla. Setite se samo njegove fantazmagorije o „bežičnom prenosu energije" s problema koje je izazvala i njemu i drugima.

Mislim dakle, da da civilizacija biti najsrećnija i najuspešnija ako Zemljinom kuglom ravnopravno budu hodili i veliki sanjari ali i skeptici jer skepsa je ponekad dobra. Ona naime spređava slobodan razmah gluposti!

Međutim, nije mali broj istraživača koji smatraju da da dođi do trajne konvergencije informativnih tehnologija – televizije telefona kompjutera i drugih – i stvaranja jedinstvenog komunikacionog sistema. Neki je opisuju kao „telematik" – kombinovanje kompjutera sa telekomunikacionim tehnologijama drugi govore o „informatiku" – računanje i informacije, uključujući i vesti treći elaboriraju „kompjunikacije" – računanje i komunikacija.

– Neke od ovih izraza sam čuo a neke ne. Sve mi se nešto čini da su njih smislili dokoni naučnici ali i novineri koji moraju popunjavati stranice svojih listova!

Nema nikakve sumnje da da u budućnosti doći do konvergencije informativnih tehnologija. Ono što je gotovo izvesno je njihova digitalizacija, što vie facti, zapravo označava njihovu kompjuterizaciju. Druga izvesnost je pojava, već i sada prisutna u najrazvijenijim zemljama all trenutno veoma skupa, komunikacionih kanala ogromne propusne moći – recimo, reda veličine terabajta u sekundi.

Ova dva elementa su sasvim dovoljna da u razumnoj budućnosti objavimo da je objedinjavanje informacionih tehnologije obavljeno – „sve ostalo su nijanse", što rekao Đoka Balašević.

Nijanse međutim mogu da budu značajne jer je unutar zdrave pameti da će se informacione tehnologije razvijati kroz komercijalnu primenu te da moćne korporacije veoma insistirati na konkurenciji i odsustvu standardizacije. To je verovatno štela, se tačke gledišta korisnika, ali je verovatno i korisno jer će stimulisati konkurenciju, pa samim tim i razvoj.

Revolucija komunikacija upravljanja informacija, revolucija informativne tehnologije kompjuterska naučna i tehnološka – u postindustrijskom svetu nije samo razvija revolucionarne načine prenošenja informacija. Pored idile „Makluanovog sela" nova realnost nudi i manipulatore diktatora profesionalne „izmišljače" onjemšane na uništavanje granice između stvarnosti i mašte. Suočavanje sa tipovima a la Bak Rodžers novim oblicima ograničavanja slobode nije više samo na polici sa delima SF produkcije

– Ne bih rekao da se radi o novim oblicima ograničavanja slobode već, ipak o novim načinima ograničavanja slobode. Razlika: na prvi pogled na deluje značajno ali mislim da je značajna se stanovišta načina „borbe protiv" jer možemo koristiti već postojeća iskustva naše civilizacija

Svaka manipulacija informacijama odnosno onim koga (dez)informišemo je uvek cilјsna – ima nalu svrhu zadatak. Najuopštenije rečeno cilј je stvaranje nekog virtuelnog sveta n kome treba da misli da živi potencijalna žrtva ove manipulacija. Način kako ak tuelni režim u Srbiji vodi informisanje preko državne televizije danas ili Crkva pogotovo hrišćanska, koja je to veoma uspešno izvodila punko hiljadu godina paradigme su ovog stvaranja virtuelnog sveta

U svakom od istorijski značajnih slučajeva koje odaberemo ja sam odabrao ovu dva jer je jedan izuzetno dugo trajao a drugi ostavlјa na svojoj koži mi možemo noći jedno trivijalno zaključiti – manipulacija informisanjem uspeva samo ako postoji monopol na informisanje

Preme toma naš hipotetični Bak Rodžers ima šansu da uspe samo ako mu mi stvorimo politički mi ja u kome će on biti jedini posednik sredstava za saopštavanje istine što nas vraća u ravan prethodnog pitanja i odgovora na njega. Da zaključim samo „checking and ballance" – ravnoteža i uzajamni nadzor kako se u američkom političkom sistemu opisuja nezavisnost sudske zakonodavne i izvršna vlasti –

Okolnost da život, terijski i principijelno, za nas više nije tajna ne mora da znači da ćemo mi, ikada, moći sa njim da „radimo šta god hocemo".

proširen i na oblast informisanja može da nas spase od potpunog ograničavanja slobode. Delimično ograničavanja će uvek biti

Život unutar sopstvenog mentalnog videa koje podržava novotehnološka stvarnost, kreira nove zadove. Današnji gledaoci sopstvenih neprekidnih filmove i sutrašnji uživaoci u usavršenim kompjuterskim bićima, sposobnim da vode ne neobuzdane mentalne putove a la „Neuromanser" Viljema Gibsona – generacija su sa kojom treba računati. Znači li to da će sa ostvariti proročanstvo Marvina Minskog – da će pisci naučne fantastike stoga što najozbiljnija pokušavaju da shvate posledice i implikacije pojedinih stvari poput Isaka Asimova i Viljama Gibsona imrot nekoliko stotina godina biti smatrani važnim filozofima XX veka a profesionalni filozofi biti načisto zaboravljeni?

– Ne verujem! Mada nije nemoguće da će savremeni vrhunski pisci naučna fantastika jednog dana biti ocenjivani povoljnije i značajnije od savremenih socio loga i psihologa

Filozofija kao pokušaj samorazumevanja je pak nešto sasvim drugo i mnogo ozbijnija

Kao što i prvi deo Vašeg pitanja sugeriše oni se ipak bava problamom virtuelnih svetova i istražuju moguće implikacije. Problam nije u njihovom odsustvu sofisticiranosti analitičke dubine ili analitičkih pratpo stavki već u pukoj okolnosti da se oni bava stražične njem hipotetičkih grananja stvarnosti. Mi sada ne znamo a ne možemo čak ni da slutimo koja će od ovih grananja postati stvarnost III deo stvarnosti koja će ostati u domenu sopstvenog „mentalnog videa" a koja će postati toliko značajna de na nas utiču ili nas čak ugroza. Teko nešto spak spada n književnost a na u filozofiju

Alvin Toffler, pisac „Šoka budućnosti" „Trećeg talasa" „Zaokreta moći" – opisuje „skankvork" (skunkwork) organizacija specifičen oblik timskog reda rasterećen birokretskih zadužanja i hijerarhijskih opterećenja. Primer okupljanja preko elektronske mreže (gde su veze između sektora ranga i geografske lokacije irelevantne) vama je blizak zar ne?

– Da sličajem profesionalnih okolnosti ali i interesovanja, stalan sam korenik i možda, ovisnik takvih mreža.

U ovom trenutku one su u povoju i predstavljaju lak jednostavniji i brži način distribuiranja inače stan dardnih informacija koje bi se mogle zamisliti i na papiru. Osim toga, nama koji smo profesionalci one omogućuju dostup odsečenim bazama podataka i kompjuterskim resursima po svetu što za širu publiku uopšte nija interesantno

Međutim mada ne znam kako će to stvarne izgledati ove mreža otvaraju neslućene mogućnosti promene tipa informisanje pa samim tim i načina života.

Redovna konferencija koja se vode na njima redovno omogućavaju ljudima koji se uopšte nzajamno ne poznaju nil jposlopi šansa da bi ikada stupili u kontakt – geografska klasne i intelektualne razlike među njima su prevelike – da razmenjuju mišljenja i što je možda važnije alternativne informacije

Recimo tokom ovog rata komunikacija se Zagrebom Ljubljanom i našim i njihovim političkim i ekonomskim emigrantima po svetu nije ni jednog trenutka bila prekidana imam nbsak da smo se mi koji obitavamo po tim mrežama mnogo bolje razumeli iako mamo suprotstavljena interese nego i najvešti i najpomirljiviji politički pregovarači na obe strane

Nova metode skladištenja ogromnih količina digitalnih informacija prolaze su svoj put kroz istraživačke laboratorije jureći prema komercijalnoj pa čak i potrošačkoj primeni. Život sabijen u kalrdž daje novi „prostor za kretanje" Svetovi privida simulacije donose na ekrane novu stvarnost. Moram da vas pitam iskorištite li i koliko iz ovozamišljenog i praktičnog u uzbudljivo i nemoguće? Igrate li se?

– Još uvek sam sklonji klasičnim vidovima „iskorištenja" iz stvarnosti kao što su literature muzika ili film i televizija. Istina u domenu vizuelne prezentacije film i televizija su veoma kontaminirani kompjuterizacijom ta mi se tako kompjuteri vraćaju „na mala vrata" u moj virtuelni svet

Ipak verujem da je to tip senzacije koja se ne mogu zaobići ali će samo mladim generacijama koja su as njima rasle značiti i nerazdvojni deo života

Što se igre n užem smisln reči tiče obožavam kompjuterske igre ali i tima cu razočarati mnoge ljubitelja sofisticiranih igara koje tragu saitima i danima mnogo sam sklonji igrama a la Tetris mada priznajem da se lako može braniti teza ba ona prevashveno služe abreagovanju neuroza, da su nešto poput štrikanja

Virtuelna stvarnost medijum je u povoju. Neće proći još dugo vremena a tehnika i tehnologije virtuelne stvarnosti predstavljaće standard zabavnog posluvnog i komunikacionog okruženja. Bliski susret sa „prisutvom na daljinu" podrazumeva kombinaciju velikog broja „interfejsa" Koliko domaći ambijent takav laskav je ima šanse da se razvija u tom pravcu?

– Pa kao i u svim dosadanjim industrijskim tehnološkim i političkim „revolucijama" – mi možemo očekivati da budemo lak recipijent nečega što je stvoreno van nas nezavisno od nas i na za nas. Mislim da u toma i leži jedan od glavnih uzroka naše kasnoće, naše nesposobnosb da se uklopimo u svet. Napoznalo nam se čini neprijatjskim. Ovo je naravno opšta mesto jer se može pripisati svakom primitivnom narodu ali to na znači da mi treba da budemo srećni i zadovolјni zbog toga što smo primitivni

Međutim u svakoj nesreći ima i neke pouke prilika za boljitak. Iz ove nesreće – informatičke revolucije – koja nas upravo „snalazi" pruža namse prilika da izvučemo korist. Nalme glodelnost promena i njihova brzina učiniće naše kaiofiicirane sociplne močnike smešnim i trapavim lakše će se sasušti i otpasti a mi ćemo lakše uhvatiti korak sa svetom. Ne zato što to želimo – mi se tome želeško opiramo – već će nas svet ne to naterati. Oni više nemaju vremana!

Vaš pogled u 1993

– Za 1993. je jedino izvesno da će nastupiti

Sva ostalo je u domenu nagađanja. Bilo bi ti cinično ili nenkusno da u ovom užasu koji nas okružuje izjavljujem ma šta.

Da ovaj moj odgovori ipak ne deluja previše turobno evo i jedna optimistična slike. Ovaj mini razgovor kada se pogledaju teme koja smo dotakli našem obaćnom čovaku čak i ako je intelektualac a razuma ono o čemu smo govonli deluje kao razgovori ludaka. Veleo bih da nam se u 1993 dogodi ne narod dosta nam ga je bilo nego vreme u kome ovaj intervju naća bati razgovor ludaka

Nastavak sa strane 20

Raznolikost tehnologije daje šansu Intelovom konkurentima da postignu uspeh se procesorima koji koriste otvaranje savrim novih segmenata tržišta kao i rupe u Intelovoj proizvodnoj liniji. Na primer, AMD je uspeo se svojim 40 MHz 386DX čipom jer je odgovorio na potrebu za 386-ticom visokih performanse. Intel je istanut iz pojedinih segmenata tržišta a tek treba da ostvari ozbiljno nporište na tržištu „ručnih" kompjutera. Međutim Intel je udružio snage sa kompanijom „VLSI Technology", u naman da proizvede čip-set klase 386SL za „ručne" mašine

Intel je spreman da ponudi čak i čipove prilagođene specifičnim aplikacijama. Njihov *RapidCad* koprocesor, lansiran u februaru prošle godine jeste dograđnja u vidn dva čipa koji odgovaraju 386 i 387 uticama. Uticnicu 386 zauzima čip koji predstavlje hibrid procesora 386 i 387 dok uticnicu 387 CPU koji se radi po narudžbi (custom) i nukuje tauzecima (tj. greškama tipa „sleljenje sa 0"). Ovaj čip-set radi na bilo kojoj 386 mašini de 33 MHz i daje performanse bolje za oko 20-30% kod izrazito grafičkih aplikacija kao što je CAD

SLEDEĆA GENERACIJA

Mada Intel nije nameravao da lansira P5 pre početka 1993. mnogi detlji o ovom mikroprocesoru već su obelodanjeni. Imaće superskalarnu arhitekturu što znači da će imati dve ili više izvršnih jednica. To će omogućiti procesoru da kotnn jednog ciklusa paralelno izvršava više instrukcija

Inidžer jednica čini paplјajn od pet nivoa čipa 486 još bržim tako što dva rade paralelno. Paplјajn dakodira dve instrukcije istovremeno pa ako nisu konfliktne izvršavajn se paralalno. Ovaj dvostruki paplјajn može za 50% ubrzati čip i anlјaj dizajna. Dva druga ključna dela inidžer jednice su *branch target buffer* (cilјni balaz za instrukcije grananja – „branch") i keš sa dvostrukim prestupom. *Branch target buffer* predviđa pojavu ogranka tako je predviđanje tačne ogranak se izvršava bez zastoja. Keš sa dvostrukim prestupom prima i podatke i adrese iz paјpova i sadrži logikn za rešavanja zavisnosti adresa.

Matematički koprocesor čipa P5 ima in posvećene aritmatičke jedinice i paplajn od osam nivoa koji je integrisan sa inidžer paplajnom ali ima još dva izvršna nivoa. Mada je matematički koprocesor podešen za dvostruko precizne *memory-to-register* operacije (najčešće očekivani tip) Intel tvrdi da se jednostruko precizne i *register to-register* operacije obavljaju istom brzinom

486 koneti jednomikronsku tehnologiju da bi smestio 1 2 miliona tranzistore. P5 koristi 0 8-mikronsku separaciju i ima 3 miliona tranzistora. Intelovi ljudi kažu da će prva verzija čipa P5 raditi na 66 MHz i preko 100 MIPS-a. Izrazito grafičke aplikacija. server se ba zama podataka za klijent/server sistema i multitasking aplikacije samo su neki od korisnika koji će na najbolji način iskoristiti njegovu snagu

Ali Intel nije jedini proizvođač koji ide dalje od 486-ice (vidi tekst „Novi čipovi ne horizontu"). Sutonih Toprani direktor marketinga i inžinjeringa personalnih računara kompanija AMD kaže da „mnaju značajne planova za čipove koji će prevazići 486". Toprani ali će da je AMD vodeći snabdevač proizvodima RISC arhitektura što je vrlo značajno s obzirom da će RISC tehnologija biti ključni faktor u sledećoj generaciji čipova. Toprani kaže da će sledeću generaciju čipova ponuditi tokom 1993

Cyrix se svoje strane najavljuje da će odmah nakon pojave P5 ponuditi svoj P5-kompatibilni procesor dok će novn generaciju čipova takozvanu klasu 686 izbaciti pre Intela. To će uspeti zahvaljujući svojim kraćim proizvodnim ciklusima. Sieljer iz *Microprocessor Report*-a kaže „Gotovo svi su potcenili tešnoće koje će se javiјati pri integraciji ovih superskalarnih čipova. Dok ne vidimo „Cyrix-ove" čipove svakve izjave moramo primati s rezervom "

Bilo kako sito mnoštvo različitih mikroprocesora stvara nedoumice pri izboru sistema. Međutim iz nevolja je potpuno zanemarive u odnosn na drastično unapređenje performanst i pojeftinjenje sistema koje je donela konkurencija među proizvođačima mikroprocesora. „Upotreba kompjutera postaje sve jednostavnija," kaže jedan predstavnik kompanije „Chemical Bank". „Što budu imali vaću moć, to će ih ljudi više koristiti."

Endi Redfern, Byte

Prevela Ranka Jovanović

TELEFAX
(021)616-887

TELEFONI
(021)623-928
624-501

Novi Sad
Bulevar 23. oktobra 25
KONSALTING I INŽINJERING

ASYS COMMERCE

EPSON

- LX 400
- FX 1050
- DFX 5000
- LQ 100
- LQ 570
- LQ 870
- LQ 1070
- LQ 1170
- LQ 2550
- DLQ 2000

SERVIS

DODATNA OPREMA

REZERVNI DELOVI

KDLIČINSKI POPUST

ZAŠTITA

- HW KLJUČEVI /HAROLOCK/
- ANTIVIRUSNI PAKETI
- SREDSTVA ZA KONTROLU PRISTUPA
- KRIPTO ZAŠTITA
- ZAŠTITA PRI PRENOSU PODATAKA
- STRUČNI SAVETI
  - ZA REŠAVANJE SISTEMA ZAŠTITE
  - ZA IZBOR OPREME
  - ZA OPTIMALNO KORIŠĆENJE POSTOJEĆE OPREME

UPS

- 300 VA
- 550 VA
- 1000 VA
- ON LINE UPS /prema zahtevu/

ASYS COMMERCE

ADRESA: GRADSKI PARK B.B. (ZGRADA CRVENOG KRSTA), 11080 ZEMUN, TEL./FAX: 36 11 613 822
PREDSTAVNIŠTVO U BUDIMPEŠTI: ANDRÁSSY UT 122, 1062 BUDAPEST, HUNGARY, TEL./FAX: 36 1 132 4923

PERSONALNI RAČUNARI • ŠTAMPAČI • PRATEĆA OPREMA

QUATTRO... PA SVI OSTALI

Pre nekoliko brojeva prikazali smo, u najkraćim crtama, *Quattro Pro for Windows*. Tada smo imali na raspolaganju samo „beta release", a sada imamo kompletan program. S obzirom da se radi o zaista „vrućoj stvari", ponovo pišemo o istom paketu. Pranaćemo vam neka nove utiske atećene u radu uz korišćenje kompletna dokumentacija, a ponovićemo i nešto od prvih utisaka koji se nisu promenili.

Vladimir Stamenović

Program se isporučuje u, za Borlandove proizvode standardno ogromnoj kutiji sa gomilom dokumentacija i pet disketa od 5 25", 1 2 Mb. Od uputstava za rad tu su:

- *User's Guide*, korisnički vodič, kompletno uputstvo za rad sa tabelom na 450 strana
- *Getting Started*, kratko uputstvo o instalaciji i mogućnostima programa (90 strana)
- *Database Desktop Guide*, uputstvo za rad sa bazom podataka (150 strana)
- *Building Spreadsheet Applications*, knjiga sa kompletnim pregledom funkcija, makroa i njihovom upotrebom u izgradnji konsničkog interfejsa na 460 strana
- *Quick Reference Guide*, kratki pregled ikona, komandi, funkcija i makroa a kojima raspolaže QPW

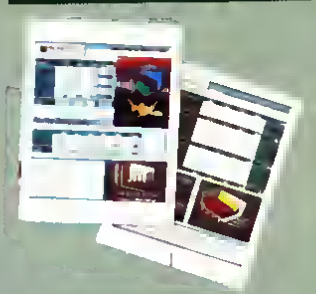
Odmah treba reći da ja QPW, kao i većina drugih *Windows* programa ima velika zahteve u pogledu hardvera. Nadao sam se da tako neće biti u ovoj finalnoj varziji- ali ipak treba obezbediti računar sa više od 4 Mb RAM-a, dok je na disku neophodno oko 10 Mb slobodnog prostora. U suprotnom, od instalacije nema ništa. Interesantno je da nakon instalacija QPW zauzima, ukupno, manje od 8 Mb prostora na disku. Onaj zahtsvani višak od oko 2 5 Mb stavljam na dušu instalacionom programu. On ja urađen veoma atraktivno, pokreće se iz *Windowsa*, ali je prilično spor, mada brži od beta-verzija. Da bi raspakovao i prabacio datoteke sa četiri disketa na disk potrebno mu je više od 35 minuta a može vam se dasti da na samom kraju instalacioni program prijavi naku grešku, ko zna iz kojih razloga, pa ćeta morati sve ispočetka.

Moram da kažem da su mi veliku pomoć pružile gospoda iz firme „Mladost" Holding – Loznica, Beograd, koji su mi, za ovu priliku, dali na upotrebu računar 386 SX sa taktom od 25 MHz, 8 Mb RAM-a, VGA karticom i monitorom, i diskom od 220 Mb. *Windows* i svi programi za njih su radili kao sat. Odličan računar.

NAJLEPŠI NA SVETU

Nakon završena instalacije možete pokrenuti program. Kao što sam u prethodnom članku napomenuo, a mišljenja nisam promanio nema lepšeg i atraktivnijeg interfejsa u svetu *spredsheet* programa. Ponoviću naka svoja osnovne utiska- jer to ovaj program zaslužuje.

Srediănji i najveći deo ekrana, razumljivo, zauzima tabela. Ona se sada zove **Notebook**, a svaka ima svoj naziv, kao NOTEBK1 WB1, NQTEBK2 W81 itd. Što na amata da tabelu snimite pod nakim svojim imenom. Možda je najzanimljivija novost u programu da je prihvaćen Lotusov koncept trodimanzionalna tabele (iz verzija 3 xx). Naime, u QPW svaki **Notebook** ima do 256 tabela i jednu grafičku stranu. Sve to zajedno se smešta u jednu datoteku! Vrlo ja lspo rešenja, u interfejsu, za promenu aktivne tabele. Ona su poredana jedna iza druge, a svaka ima izrez, kao u rokovniku, obeležan nazivom tabele (od A do IV, a možete dati svakoj strani i svoja sopstveno ime).



Vidljivo je samo sedam prvih naziva tabela, dok za ostala postoji, u donjem levom uglu, mali skrol-bar pomoću koga se može doći i do ostalih tabela. Brzi prelaz iz aktivna tabele u poslednju i obrnuto se ostvaruje klikom na **SpeedTab** dugma koje se nalazi u sredini donjeg reda na akranu, desno od oznaka tabela, a obeležano je strelicom levo ili desno, zavisno od toga na koju stranu se kreće. Za prelaz iz jedna u drugu tabelu, dovoljno je da mišem kliknata na odgovarajuću oznaku. U levoj polovini donjeg rada ispod tabele i sa leve strana tabele, nalaze se horizontalni i vertikalni skrolbarovi, koji, kao i kod svih *Windows* programa, služe za kretanje po tabeli.

Iznad tabele, u prvom redu, nalaze se komande za rad sa programom. Njihov raspored, aktiviranja, kao i skraćenice za rad su uobičajani u ovakvoj vrsti programa. Prva po redu je **File** komanda kojom se barata sa datotekama, štampaju dokumenti i izlazi iz programa. Sledeća je **Edit** komanda, koja omogućuja rad se **Clipboard**-om (**Cut, Copy, Pasta**), a sadrži i komanda **Goto, Undo** i **Search**. Za reararanžiranje tabele služi komanda **Block** (**Move, Copy, Fill, Reformat, Insert** i **Delete** kolona, redova i strana). Rad sa bazom podataka se odvija kroz komandu **Data**, a veoma zanimljive mogućnosti nudi komanda **Tools**. U njen domen spadaju rad sa makroima, kombinovanje i linkovanja datoteka, rad sa matricama, „šta-ako" analiza. Solvar i mogućnost promene izgleda meni-bara (*UI building*). Grafičku prezentaciju podataka ćete raditi kroz komandu **Graph**. Tu možete kreirati doterivati i posmatrati grafikona i izv. *Slide show*, kao i ubacivati gotova alličice (*clip art*). Komandom **Property** možete vršiti razna podešavanja za svaku vrstu objakta, dok za reararanžiranje prozora služi komanda **Window**. Na kraju menija ja komanda **Halp**.

Ispod menija nalazi se „bar" sa nizom različitih ikona koja bi trebalo da ubrzaju najčešće obavljana poslove u tabeli (pun naziv je **Notebook Window SpeedBar**). Ima ih dvadesetak. Prve tri služe umesto komandi **Cut, Copy** i **Paste**. Podaci iz ćelija ili blokova se vrlo brzo, pomoću ovih ikona, mogu isecati, kopirati i premeštati bilo gde u tabeli, u sve tri dimenzija. Tu je i ikona za ubacivanje simbola za dugmad kojima će se izvršavati neka makro procedura. Brzo kreiranja grafikona, poravnavanje teksta na sva tri načina, promana formata teksta (**bold** i *italics*) i brojčanih podataka je samo deo posiova koji su znatno ubrzani preko ikona. Postoji i ikona koja omogućuje brzo sabiranja promenu visina reda ili, vrlo interesantna, **SpeedFormat** ikona koja otvara dijalog-boks sa pet unapred definisanih formata za tabelu. To znači da je već određeno kako će tabela izgledati, sa isertanim linijama ispod zaglavlja, osenčenim važnijim blokovima, različitim tipovima fontova... Impresivno! Treba reći da se „bar" sa ikonama menja u zavisnosti od posla koji se radi. Recimo, ako adtujete grafikon, tu su potpuno nove ikone koje omogućuju veoma komforan i lagan rad na doterivanju izgleda grafikona.

U trećoj liniji ekrana je smešten „edit bar" (**Edit Mode SpeedBar**). U njegovom levom delu se vidi adresa aktivne ćelija ili bloka, a ako pritisnete taster **F2**, onda se u nastavku vidi i sadržaj koji se može korigovati. Pri unosu podataka od pomoći su ikone koje ubrzavaju posao oko, na primer, ispisivanja makro komandi (zajedno sa vitičastim zagradama).

Borland-ovi programeri su učinili veliki napor da korisnicima omoguće komforan i brz rad i u tome su u potpunosti uspeli.

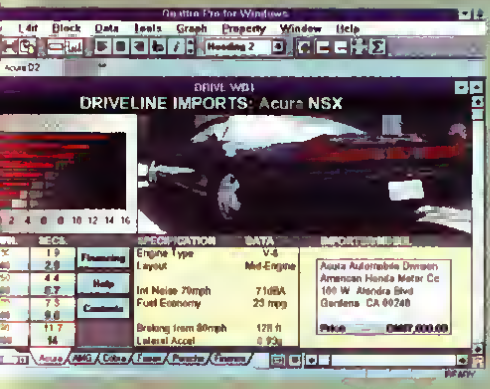
S obzirom da ja ova verzija QPW instalirana na dobrom 386 računaru, program je radio veoma brzo. Ni traga od nepoželjnih i sporih osvežavanja ekrana. Ikona su brzo odgovarale na aktiviranje, pa je rad pravo zadovoljstvo.

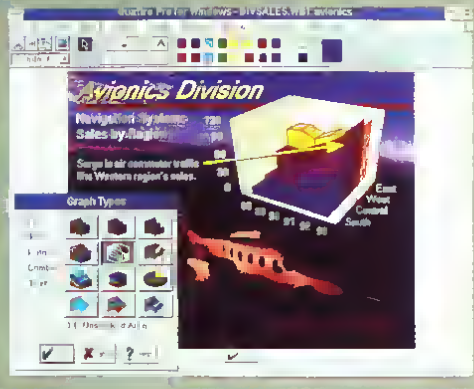
Dokumentacije ja, po običaju, sasvim zadovoljavajuća i pomoću nje se mogu relativno brzo savladati osnove programa. Help je takoođe na zadovoljavajućem nivou i iz njega se mogu saznati sve ili gotovo sve osnovne stvari o programu.

RADNI PROSTORI

QPW ima pet radnih prostora. Prvi je **Notebook Window**, prostor gde unosimo i prikazujemo podatke, a mi ga nazivamo tabela. Sadrži sve standardna mogućnosti *spreadsheet* programa, a naravno i mnoga proširenja i dodatka. U njemu se obavlje najveći deo posla. Kreiranje, doterivanja i isprsvka grafikona, bazirania podacima iz tadela se obavlja u **Graph Window**. Ako pravite sopstvenu aplikaciju koja će raditi u QPW, konstićeta **Dialog Window Grapha Page** je poslednja strana u svakoj tabeli (**Notebook**). Tu vršite povezivanje grafikona i slika u *slida show*, ili povezujete sopstvena dijalog- boksove koji će činiti naku aplikaciju. Peti radni prostor je **Print Preview Screen**, gda sa može pregledati rad pre štampanja.

Da se vratimo tabeli. Dakle, kao što sam rekao, imamo trodimenzionalnu tabelu. Svako ko je lkaoa radio sa Lotusovim 1-2-3 verzija 3 xx zna kakva je to prednost u odnosu na „običnu" tabelu. Primera radi, ako imate nekoli-





ko tabela koje služe za obavljanje sličnih poslova ili su, čak, međusobno linkovane, onda je daleko čistije situacija ako svaku od tih tabela ubacite u poseban list u jednom **Notebook**-u. Tu je povezivanje podataka daleko lakše, a rad znatno brži. Na kraju krajeva, umesto nekoliko datoteka imaćete samo jednu. Poslednja strana u **Notebook**-u je grafička strana koje sadrži ikone koje predstavljaju svaki od napravljenih grafikona, *slide show* ili dijalog boksove koje ste sami kreirali. Karakteristično za QPW je da skoro svi elementi koji se vide na ekranu predstavljaju objekte. Svaki od njih se može posebno podešavati i doterivati. Na primer, u objektu koji se zove *Aktivni blok*, možete vršiti najrazličitija podešavanja na aktivnoj ćeliji ili bloku, ili to isto, ali na **Graph** objektu.

Kao u verziji QP 4.0 za DOS, i ovde mi sa najviše dopalo da radim u grafičkom prozoru. Pravo je zadovoljstvo kreiranje i doterivanje grafikona, mešanje sa gotovim sličicama (probao sam samo CGM format) ili njihovo preuređenje. Ovaj deo programa je urađen vrlo kvalitetno, se dosta mašte i pažnje. Kada editujete grafikon ili sliku, u posebnom prozoru, u **SpeedBar**-u dobijete nov set ikona, a meni ima nekoliko novih komandi. Uglavnom posao je jednostavan. Klikom obeležite objekt, a zatim preko ikona ili kroz meni vršite odgovarajuće doterivanje. Dijalog-boksovi su impresivni, na višem nivou od onih u *MS Excel*-u. U njima sa vidi reakcija na svaku promenu u podešavanjima nekog objekta. Posebno ma je zainteresovala mogućnost da preprsvijam gotova sličice. Načini na koji se to može urasditi gotovo da nemaju broja: promena boja i nijansi svih elemenata na slici, promena šrafure i punoće boje, debljina linija, dodavanje novih elemenata, itd. Vrlo lako i zabavno.

QPW je poboljšan u više pravaca u odnosu na proizvode za DOS. Recimo, ima dosta novih funkcija, tako da ih je sada ukupno 121. Izdeljene su u nekoliko grupa: matematičke, statističke, *database*, logičke, finansijske, za datume i vreme, i string funkcije. Neme sumnje da će ovoliki broj dobro osmišljenih funkcija omogućiti korisnicima da prave dobre i kompleksne tabele za najrazličitije poslove.

Što se tiče kompatibilnosti se 1-2-3, izgleda da je sve u radu i da QPW prihvata sve makroe i funkcije iz 1-2-3 verzije 2.xx i 3.xx. To bi trebalo da bude veliko olakšanje za one korisnike programa 1-2-3 koji se, eventualno, reše da pređu na rad se QPW. Prilikom ubacivanja makroa iz Lotusovih tabela ne treba zaboraviti da se mora aktivirati takozvani „1-2-3 menu tree", a za upotrebu uobičajenih komandi u 1-2-3 i QP za DOS treba aktivirati **Key Rider**. Makroi koji se u 1-2-3 pokreću uz pomoć **Alt** tastera, ovde će se startovati pomoću **Ctrl** tastera.

U ovoj verziji, QPW ima zadivljujućih 150 makro komandi izgleda da su Borlandovi programeri mislili na sve, jer, prosto nema onoga posla koji se pomoću njih ne može uraditi. Ovo ocenjujem kao veliki napredak u *spreadsheet* programima. Kod pisanja makro procedura od velike pomoći je ugrađeni **Macro Debugger**, pomoću koga se makroi mogu kontrolisati na nekoliko načina: korak po korak, postavljanjem tačaka prekida, kada makro staje ako je zadovoljo određeni rezultat ili uslov, pokretanjem makroa punom brzinom do tačke prekida, a onda korak po korak. Rad se dibegerom se odvija u posebnom prozoru – **Debug window**.

POTPUNA PRILAGODLJIVOST

Veliki kvalitet programa QPW je to što ga korisnik može sebi maksimalno prilagoditi. Recimo, na morate koristiti standardne *Windows* meni komande, već to možete raditi iz nekog od alternativnih (1-2-3 ili QP) menija koje ćete pozivati sa „/" „Meni-bar" možete zamenili sopstvenim menijem, koji će sadržati vama potrebne komande. Za to treba iskoristiti mogućnosti koje se nude u alatima za izradu sopstvenih aplikacija. U ovom programu aplikacija je mešavina i kombinacija makroa, dijalog-boksova i menija. Za izradu aplikacija postoje alati, takozvani **Application Tools**. Pomoću njih možete doterati *Quattro Pro* tako da lapunjava specijalne zahteve korisnika, možete isključiti nepotrebne komande, čime interfejs postaje jednostavniji (setite se „Balera" za 1-2-3), a postoji i mogućnost kreiranja interfejsa za nove, vaše komande.

Taj posao, pravljenje aplikacija, predstavlja pravo zadovoljstvo. Napravićete sopstvene dijalog boksove, **SpeedBar**-ove i menije i veoma ih jednostavno povezati u celinu – aplikaciju. Sve sa odvija u dijalog-prozoru do koga se dolazi komandama **Tools Builder**. Kada se nalazite u njemu, imate na raspolaganju poseban **SpeedBar** se potrebnim alatima. Vaš dijalog-boks može sadržati razne oznake, polja za unos podataka, takozvane radio tastere koji će aktivirati različita podešavanja, list-boks sa listom opcija ili bitmap tastera (obično **OK** i **Cancel**). Povezivanje dijalog-boksa sa aplikacijom ide preko makro komande <DODIALOG> se odgovarajućim argumentima.

Za poslove koji se često izvode, korisno je napraviti sopstveni **SpeedBar** koji će sadržati skraćenice za te poslove i tako ih znatno ubrzati. Razlika između **SpeedBar**-a i *dijalog-boksa je u tome što se SpeedBar pojavljuje na vrhu Quattro Pro prozora* i konenik ga na može ukloniti (osim ako to nije makroom predviđeno) i čuva se u odvojenoj BAR datoteci. Da bi ste ga prikazali u aplikaciji, morate prvo ukloniti originalni, a onda, postaviti sopstveni **SpeedBar**.

Što se menje tiče, možete praviti sopstvena menije pomoću čijih komandi ćete inicirati ikone, pokretati makroe, prikazivati dijalog-boksove ili pozivati druge menije. Naravno, sve će to biti u *Windows* stilu.

Uputstvo „Building Spreadsheet Aplications" je prilično solidno, pa mislim da korisnik ne bi trebalo da ima većih problema u pravljenju aplikacije.

Ovo je, po mojoj oceni, najbolji spreadsheet koji radi pod *Windows*-om. Ostavio je na mene bolji utisak od *MS Excel*-a, da i ne govorimo o programu 1-2-3 za *Windowse*. QPW ima fantastičan interfejs, manipulacija objektima je izuzetna, kretanje po trodimenzionalnoj tabeli je veoma lako, a ovoliki broj i ovakve funkcije i makro komande te sistem gradenja aplikacija, garantuju da će korisnici imati velike mogućnosti u pravljenju sopstvenih aplikacija. Ako vam ne smetaju popnični zahtevi u pogledu hardvera (min 4Mb RAM-a i gotovo obavezni 386 procesor), *Quattro Pro for Windows* će sigurno zadovoljiti vaše potrebe a ponavljam – mislim da je jedan od najboljih, ako ne i najbolji *Windows spreadsheet* na tržištu. No, videćemo da li će ovaj program, bez velike DOS baze korisnika, uspeti da sa nametne kao standard.

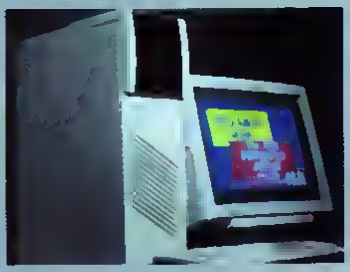
Korisna adresa:

SOFTLAND
11000 Beograd, Skadarska 45
Tel 011 / 343 043

Sve standardne
konfiguracije
PC računara

LapTop i
notebook
računari
Veridata

ALR
Advanced Logic Research Inc



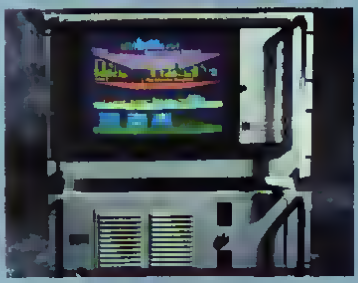
Snaga i sigurnost
koja Vam je potrebna

Računarske mreže pod
operativnom sistemom **NOVELL**

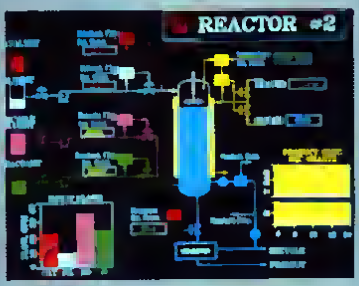
UNIX, XENIX
Open Desktop
SCO
THE SANTA CRUZ OPERATION

Kompletno rešenje za automatizaciju industrijskih procesa i laboratorija bazirano na PC računarima

PC-LabCard / Industrijski PC računari / PC-ToolCard



TEXAS
MICRO



COMTEC

ADVANTECH

LABTECH

"COMTEC" d.o.o. Preduzeće za informatički inženjering i automatizaciju
Novi Sad, Vojvođanskih brigada 7/IX, tel:021/27-181, tel/fax:021/22-258

POČETAK I KRAJ JEDNE GODINE

INTERSOFT

29. novembra 43/I, tel: 323-539, 325-393, fax: 324-120

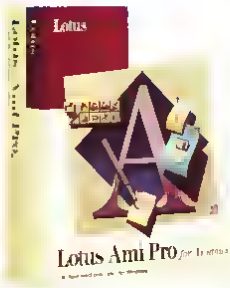
NAJBOLJI PIŠČEV PIJATELJ

„Lotus Development Corporation" je velika firma. Njihov spredšit *Lotus 1-2-3* je jedan od najuspešnijih programe ikade napisan za PC, *Freelence Grephice* jeden od nejmoćnijih peketa za poelovnu grafiku, dok je *CC:Meil* vodeći avetski LAN sietem za elektroneku poštu. Ovom impozantnom ekupu odnedavno se pridružio i *Ami Pro*, verzije 3.0, koji pretenduje da poetane tekst procesor broj 1 za *Windows*.

Ante Čurlin

Sa osobitim zadovoljstvom prilazim pisanju ovog prikaza – razlog bih mogao da potražim, pre svega, u višegodišnjem traganju (za izgubljenim vremenom?) za idealnim tekst procesorom. Ukratko, mislim da sam pronašao Sveti Grel tekst procesora, i koliko god da sam subjektivan, bez namera da povredim bilo koga – „vordovce", „vordperfektovce", ili korisnike „Chi Writer"-a i drugih tekst procesora – trudiću sa da svaku svoju tvrdnju stavim na, kako kaže Klé, „probni kamen činjenica".

Konkurencija na polju tekst procesora je u svetu veoma oštra, rekao bih čak – nemilosrdna. Svako zaostajanje, bilo u argonomiji bilo u performansama, skupo se plaća. Na takvom tržištu među divovima kao što su „WordPerfect" i „Microsoft", svoja mesto je odlučio da izbori i Lotus – se pravom. Čitajući strane časopise i razgovarajući sa ljudima (se univerziteta Iowa, UCLA, New York), saznao sam da svi pralaze na *Ami Pro*. U ovom prikazu, pokušaću da odgovorim na pitanja: Zašto?



TRIJUMF ERGONOMIJE

Ami Pro 3.0 je uraden savršeno – potpuni trijumf argonomija, performansi (brzina reda modularnost, „razumevanja" sa velikim brojem programa drugih proizvodača (na samo tekst procesorel), rad sa grafikom, red u mreži...) i cena.

Ami Pro 3.0 traži berem 286 procesor i 2 MB RAM memorije. Moraćete da pražalite cca 15 MB na hard disku (u punoj konfiguraciji). Inače, moguće je instafirati *Ami Pro* i u minimalnoj konfiguracili, na približno 5 MB. Računajte na dodatnih 3 MB na hard disku za „temporary" fajlove, plus prostor za vaše radna dateteke. Namojte da vse ovo uplaši – *Ami Pro* je vradan toga! Da odmah budamo i realni – prijatan rad zahtava barem 386 mašinu i 4 MB RAM memorija.

Ami Pro varzija 3.0 se isporučuje u 3.5"/1.44 varijanti na sedam disketa, od kojih sedma pripada ATM (*Adobe Type Manager ver 2.0*) *Windows* varzlji, koji služi za optimizaciju i povećanja oštrina ekranskih i štampačkih fontova. Nećete verovati kakav ispis možete dobiti na 9-pinskom štampaču iz *Ami Pro*-a! Takvu oštrinu (i grafike i fontova) odavno nisam video. Ono što ja vrlo značajno: Lotusov *Ami Pro* je TOTALNI WYSIWYG (već i vrapci na granama znaju šta ovo znači, za ona druge – ono što vidita na monitoru, baš to ćete dobiti i na štampaču).

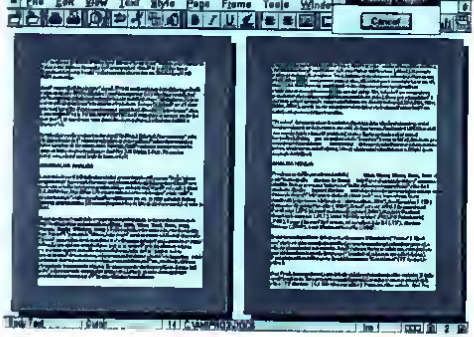
Instalacija se vrši sasvim standardno (File/Run), B install. Na samom početku sladuje pitanje da li želimo instalaciju na serveru ili na. Posla odgovarajućeg izbora, postupak ida automateki. Na kraju, *Ami Pro* otvara novu programsku grupu u koju smešta dve ikone – *Ami Pro 3.0* i *Dialog Editor*.

GENERALNA ANALIZA

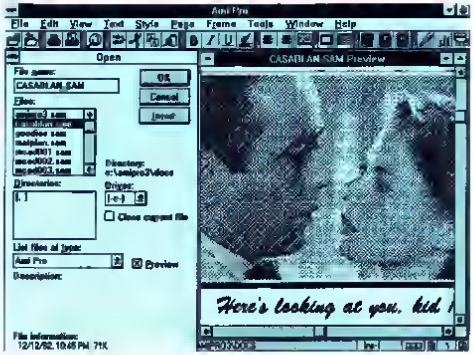
Lotusovi eksperti iz Odeljenja za tekst procesore obratili su naročitu pažnju na argonomski momenat, što smatram jednom od krucijalnih prednosti *Ami Pro*-a nad svim ostalim tekst procesorima koje sam konstio (*Word Perfect 5.1, Word 2.0, Chi Writer, Word Star*). Pretpostavimo da ste relativni početnik – odličan preduslov za početak rada u *Ami Pro*-u (mislim da će to biti početak jednog dvnog prijateliistva). Sve ono što pomislite da bi trebalo da se uradi, baš tako će i biti.

Da pogledamo stvari izbliza – u gornjem monitorskom redu sa nalaze komanda koje deluju sasvim poznato (**File, Edit, View, Text, Style, Page, Frame, Tools, Window, Help**). Već ispod njih, u drugom redu smešten je deo tzv. „smart" („pamatnih") ikona, koje drastično ubrzavaju rad. Ne morate se više „šetati" po menijima i podmenijima tražeći komandu koje vam je potrebna – dovoljan je samo klik mišem na odgovarajuću ikonu i posao je gotov! Poslednja ikona sa desna strana omogucava pozlv novih ikona, tako da ja izbor zaista impozantan. Pn svemu tome, u donjem monitorskom redu možata trenutno dakla jednom jednostapenom komandom promenti font, valičinu, razlm rada, aktivnu stranu koju trenutno obrađujete ili samo posmatrata i kontrolišeta. Za ona koji odluče da se posvete ovom tekst procesoru, Lotus je obezbedio mogućnost pravljanja sopstvenog grafičkog manlja i sopstvenih ikona.

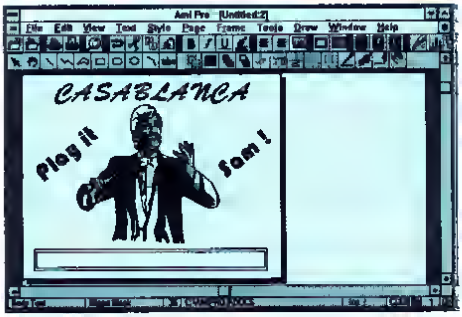
Što se tiče tipografskih karakteristika *Ami Pro* se može pohvaliti moćnim alatima, dostojnim jednog *PageMaker*-a *FramaMaker*-a ili *Ventura*. Za kontrolu radnog prostora, ili drugim rečima dokumente na kojam redimo stoja nam zaiste impozantna mogućnosti: margina, slika inkorporirana u tekst, peragrafi mreže, itd. Ono što me ja posebno obradovalo je mogućnost proizvoljnog skaliranja radnog dokumenta (njegov *default* je 91%, tako da omogu-



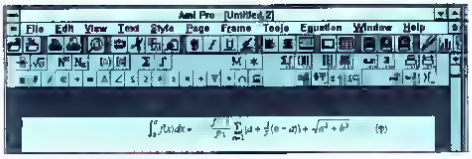
Rad sa takatom (*Facing Pages*)



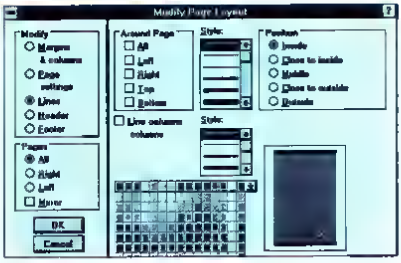
Praviaw mod kod otvaranje dokumenta



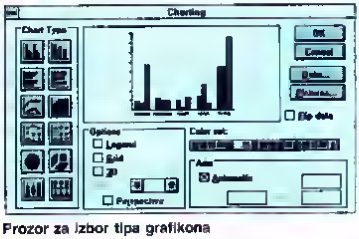
Import grafike u *Ami Pro 3.0*



Meni matematičkog editora



Podešavenja izgleda strane



Prozor za izbor tipa grafikona

čava punu kontrolu celokupnog dokumenta u vidnom polju) Da se prisetimo klasični desktop paketi omogućavaju predefinisane vrednosti skaliranja (25%, 50%, 100%, 200% 400%) a ovde imate mogućnost da odaberete baš onaj faktor koji vam u datom trenutku najviše odgovara

Rad sa paragrafima je doveden do savršenstva Paragraf čini bilo kakav tekst, slovo, ili razmak <Space>, ali između dva pritiska na taster <Enter>. Sa paragrafom u *Ami Pro*-u možete činiti sledeće stvari a) menjati font boja i atribute teksta (podebljan (*Bold*) i zakošen (*Italic*)) b) poravnavati tekst nalevo, nadesno, centrirati ga, kao i poravnavati sa obe strane (*justified*) c) postavljati tabulatore, d) uvlačiti tekst, e) određivati razmak između redova, f) prelom strana (pre, posle ili u okviru samog paragrafa) g) formatirati tekst kao bilten h) razdvajati paragrafe linijama čiju sa atributi takoda mogu menjati, i) obrađivati aritmetički podatke unutar tabela (4 osnovne operacije +,-,* / , i %), j) deliti reči na kraju reda, itd Moguća je raditi i sa dva strane na jednom dokumentu (*facing pages*) Ukratko, rad sa tekstom je olakšan do krajnjih granica, i u ogromnom broju slučajeve korisnike liševa potrebe da prelamaju tekst u nekom desktop paketu (još i njega treba naučiti!) Sve se može uraditi u *Ami Pro*-u.

„Preview" dokumenta je rešen vrlo intelligantno lako i *Amijevi* glavni supamici imaju ovaj mod *Ami* koristi UVEK korak ili dve manje da bi rešio neki problem (u tome i jeste optimizacija i pomoć korisniku) Dakle pre nego što se odlučimo da učitamo neki tajl *Ami* nam pruža mogućnosti da završimo u neotvoreni dokument tako što otvara dva prozore jedan u kojem su nazivi tajlova, i drugi, u kojem je fajl koji nas zanima

ŠETNJA PO MENIJIMA

Prva komanda **File**, pored standardnih (**New, Open, Close, Save, Save As**), sadrži novu opciju **Revert to Saved** koja vas vraća na poslednju sačuvanu verziju ukoliko vam se promene koje ste načinili u dokumentu ne svidaju i želite da ih anulirate Opcija **Import Picture** zavređuje da bude spomenuta iz prostog razloga što je spisak grafičkih formata koja direktno podržava impresivan To su *AmiDraw* (sopstvene biblioteka sa ekstenzijom SDW), *AmiEquation* (.TEX), *Computer Graphics Metafile* (.CGM), *DrawPerfect* (.WPG), *Encapsulated PostScript* (.EPS), *Lotus Freelance Graphics* (.DRW), *Hewlett-Packard Graphics Language* (.PLT), *Lotus 1-2-3 Graphics* (.PIC) *PC Paintbrush* (.PCX), *Tagged Image File Format*, od verzije 4.2 do 5.0 (.TIF), *Windows Bitmap* (.BMP). kao i *Windows Metafile* (.WMF)

Da biste uveli grafiku u dokument, slika mora biti u okviru (*frame*). Okvir možete kreirati pre unošenja grafike (prvenstveno zbog planiranja prostora), ali nije strašno eko to i zaboravite *Ami Pro* će to urediti sam, s tim što ćete morati da naknadno podešavate veličinu okvira (mišem sa to radi kao od šale) Pogledajmo Hemtri Bogarta i Ingrid Bergman u „Kazablanci" (TIF format)

Ami Pro 3.0 omogućava (pored skaliranja i pozicioniranja slike u okviru) i dalju grafičku obradu Tu se prvenstveno misli na podešavanja parametara skeniranih slika u TIF formatu (4, 6 ili 8-bitna grafika). Po unosu slike u okvir, *Ami Pro* automatski podešava osvetljenje i kontrast da bi dobio najbolje rezultate Za ručno podešavanje potrebno je aktivirati opciju **Tools/Image Processing**, i zatim *podešavati osvetljenje (Brightness)* kontrast (*Contrast*) kontrast između same slike i ivica unutar slike (*Edge Enhancement*), kao i „mekoću" slike (*Smoothing*)

Opcija **Doc Info** je standardna za „ječa" tekst procesore i nećemo se na njoj zadržavati. Vredan pomena je trud (ideja, prvanstveno!) Lotusove ekipe da ugradi u ovakav paket nešto što bi sa moglo nazveti *PC Tools* u malom. Naime, opcija **File Menagement**, bar kako je meni „zazvučala", treba da prakline i poslednju aponu sa dozlaboga dosadnim DOS-om (ili *Windows File Managerom*), za ljude koji se bave obradom teksta e ne žele uopšte da „izlaze" iz svog tekst procesora Zaista, kad sam problem počeo da posmatram iz ovog ugla shvatio sam da je konsnicima dovoljan „mini DOS servis" (**Copy, Move, Rename, Delete, Attributes**) kao i opcija **View,** da bi zadovoljili sve svoje potrebe

Master Document omogućava pravljenje sadržaja, indeksa, kontrolu numeracije strana obeležavanja i svih stvari dostojnih najkvalitetnijeg tekst procesora i desktop paketa

Opcija **Merge** omogućave automatsko povezivanje podataka iz fajla sa na primer, adresama imenima i prezimenima ljudi – ovo je specijalno interesantno kod slanja cirkularnih pisama, što *Ami Pro* (jednostavno i elegantno a kako bi drukčije?) radi automatski Svega su tri koraka u pitanju kreiranje ili preuzmite već gotov fajl sa imenima i adresama (vaših poslovnih partnera?). otkucajte cirkularno pismo i izdajte komandu **Merge** i **Print** To je sve!

Opcija **Print Envelope** omogućava da štampate adresa na koverlama Ako se u bilo kom momentu „izgubite" (pretpostavimo da je sad pravi trenutak za to) iskoristite *Ami*-jevu pomoć u kontekstu Istovremeno pritisnite taster <Shift> i funkcijski taster <F1> Strelica će se pretvoriti u neobičan znak pitanja Jednostavno kliknite na odgovarajuću komandu iz menija koja vas interesuje i dobićete detaljnu pomoć vezanu baš za tu komandu

Ami se pobrinuo da olakša i posao oko kreiranja dokumenata Čega sve tu nema – preko 50 unapred definisanih struktura od poslovnih pisama, cirkulara agendi raznoraznih kalendara (sa danima, nedeljama, mesecima) leks dokumenata sa logotipovima, profesionalnih rešenja za desktop publikacija, raznih labela, plakata EPP strukture, naučnih disertacija indeksa memoranduma molbi žalbi, pisama novinskih stubaca, zaglavlja fus-nota raznih izveštaja itd Ako želite da otkucate neki dopis, a ne znate formu odaberite iz *Ami*-jevog menija ono što vam treba – ostatak je najobičnija rutina

KONTROLA TEKSTA

Ovo je vrlo moćan deo Lotusovog tekst procesora. Sa tekstom se mogu čuda činiti, ali sve to ne zahteva veliki trud da bi se naučilo. Kako se menjaju osnovni atributi teksta? Prvo se obeleži željeni deo dokumenta (najjednostavnije i najbrže mišem – kao i kod svakog *Windows* orijentisanog tekst procesora), zatim se otvara opcija **Text/Font** i – gotovo. Možete obojiti tekst, možete odabrati proizvoljnu veličinu – uz standardne mogućnosti podebljavanje podvlačenja (čak i dvostrukom linijom !), ili pretvaranje teksta u italik.

Treba naglasiti da *Ami Pro* verzija 3.0 koristi sve *Windows True Type* fontove. Da ponovimo još jednom, kontrole veličine i tipa fonta se mnogo brže može obavljati sa donje monitorske linije *Ami*-ja – u jednom koraku.

Kontrola izgleda cele strane se vrši preko opcije **Page/Modify Page Layout**. Tabele pomeralara je detaljna i jasna – možete raditi sa tekstom bukvalno sve što hoćete.

Ne treba zaboraviti još jednu dobru ideju realizovanu kroz opciju **Text/Feet Formet**. Pretpostavimo da imate jeden dokument na koji ste utrošili silno vreme podešli ste fontove veličine, paragrafe, odje, stilove, sve što je moglo da se uradi, uradili ste. E, sad nastaju problemi – imate još nekoliko dokumenata (neobradenih, recimo u čistom ASCII formatu) i želeli biste da izgledaju isto kao i onaj prvi. Ništa lakše – izaberite opciju **Feet Formet** i sledite uputstve (iz knjige ili iz **Helpe**, potpuno je svejedno). I doći ćete do željenog cilja. Da sam pre nekoliko godina imao ovakvu aletku, ne bih se mučio sa tipskim kucanjem tekstova.

PROFESIONALNI ALATI

Opcija **Toole** krije u sebi glavne alate koji su potrebni specifičnim konsnicima. Uz već dobro poznati **Spelling Checker** tu je i **Grammar Checker**. On će vam ponuditi statistiku i ukazati na greške koje sle počinili u gramatičkom domenu naravno, engleskog jezika), dok će sa druge strane **Spelling Checker** da „odradi" svoje. Za one koji prevode na engleski jezik – prava stvar.

Rad sa tabelama predstavlje pravo uživanje – za razliku od pojedinih tekst procesora koji ovakav posao pretvaraju u noćnu moru. Jeden klik mišem na **smert** ikonu i *Ami* nudi *default* tabelu se 5 redova i 5 kolone – podesite mišem koliko vam treba i – gotovo. Rad sa tabelama uključuje elementarne aritmetičke operacije – možete sabirati sadržaje ćelije oduzimati ih množiti i delili, kao i nalazili procenel (%). Po otvaranju tabele u meniju se pojavljuje nove komanda **Teble**, koja omogućava veliki broj operacija se tabelom i promenu svih etnbula. To je toliko veliki deo *Ami Pro*-a, da sigurno zaslužuje posebnu analizu.

CRTANJE I GRAFIKONI

U *Ami Pro*-u možete nacrtati sliku (naravno, ako imate miša) koja sadrži i komplikovanije operacije (i elemente) od već legendarnog *Paintbrush*-a koji pripada *Windows*-ima. Dakle linije, lukovi, krugovi, elipse, pravougaonici i kvadrati, sve to „gamirano" sa puno unapred definisanih tekstura odja oblika, itd. Lepe je stvar što se mogu editovati Lotusovi PIC fajlovi (ako su već kreirani gratikoni, zašto ih ne iskoristiti?). *Lotus Freelence* fajlovi kao i *Drew Perfect* fajlovi – zašto baš njih spominjemo? Jednostavno zato što ih *Ami Pro* automatski konvertuje u svoj SDW format, tako da ih možete menjati do mile volje. Po izboru opcije **Tools/Drawing**, pojavljuju se nove **smert** ikone.

Pored standardnih opcija *Ami* uvodi i neke specifične, koje već pripadaju grafičkim paketima koji su samo za to namenjeni. Na primer ukoliko crtate ravnu liniju, e želite de to bude pod uglom od 45 stepeni, prvo pritisnite <Shift>, a onde pokrećite miša. Zatim, tu je interesantna mogućnost rotacije teksta za proizvoljan broj stepeni (pri tome se originalni font zadržavaei), zatim grupisanje i rotacija delinisanih objekata, puna kontrola odje mreže, itd. Ukratko dobili sle moćan crtački alat pomoću kojeg možete nacrtati ono što vam treba, preuzeti već gotovu gratiku iz celokupne game grefičkih formata ili doradili već kreirenu gratiku. Ako je i od jednog teksl procesora više je nego dovoljno!

Ono što je pre nekoliko godina bila privilegija Lotusa (paketa 1-2-3 i njemu sličnih), seda je semo ugrađeni deo tekst procesora. Ala se vremena menjaju! Dakle, podržani su svi tipovi grafikona: trekasti, kumulativni linijski površinski kombinovani linijski sa slikom gralikon u obliku pite i mnogobrojne kombinacije čije bi prevođenje moglo da izazove semo smah kod čitalaca – nazovimo ih kombinovani grafikoni. Ime tu svega i svačega – od postavljanje mreža, komentara fusnole, legendi, do odredivanje perspektive. Da se ne bismo previše udaljavali od tekst procesora (jer, ipak, o njemu je reč), i de se ne bih osećao kao da pišem o Lotusu ili Excelu, preći ćemo na drugu temu.

MATEMATIČKI EDITOR

Kao konsnik matematičkih pakete, osećam se pozvanim (i prozvanim) da analiziram malo pobliže *Ami*-jev matematički editor. De ne bi bilo zebune provente prvo da vam slučajno neko nije obnsao *Windows* kompstibilni *Symodi font* (za prikazivanje i štampu jednačina). Ako je to u redu, otvorimo meni **Tools/Equetione**. U ovom modu mozemo ubacivati „lemplate" u jednačine, kao i sve poznate matematičke simodle, funkcije itd. Pošlo moje kolege uglavnom rade u *TeX*-u obradovaće ih da *Ami Pro* prihveta i *TeX* komandel. Naravno, moguće je sačuvati jednačinu ili neki njen deo u *TeX* formatu, kao i importovati *TeX* fajl. S obzirom da je *TeX* prihvačen kao univerzitetski standard ze matematički slog, evo još jedne važne spone. Izmedu (ipak isuviše komplikovanog *TeX*-a) i jednog krajnje „user-trendly" orijentisanog pakela.

Test koji sam mogao da izvedem za relativno kratko vreme je tesl zveni „ruska matematičke zbirke". Priznajem da sam izgubio bitku – nisam pronešao matematički izraz ili simbol koji *Ami* ne bi mogao da prikaže (naravno to ne znači da ih nema). Fasciniren sam brojem opcija, bibliotekom matematičkih simbola (operatora, matrica, integrala, suma, proizvoda itd.), ali perfekcije ide dotle da možete da birate da li hoćete indeka levo ili desno, gore ili dole, podebljan ili nepodebljan simodi, u ovoj ili onoj odji i tome slično – da se čovek jednostavno zapita: hoće li mi to ikeda zatrebati? S obzirom na ranije palnje zbog skromnih mogućnosti takst procesora (a bogami, i krajnje mračnog Interfejse *Ventura 2.0* za pisanje jednačina) ovo mu dođe kao melem na matematičku ranu.

Meda sam oduševljen *Ami*-jevim matematičkim editorom, moram da priznam da mi sa učinio pomalo sporim na mojoj 386/20 mašini (ili je, ipek, *Windows* okruženje u pitanju ?). Možda bi 486/66 se lokal basom maličice poprevio stvar?

RAZMENA PODATAKA

Zbog velikog ličnog rekustva, ovom aspektu pridajem posebnu pažnju – vrednost paketa procenjujem (naravno, ne bezuslovno!) i kroz komunikaciju sa drugim (standardizovanim) pakelima. Iako cenim „Microsoft" moram da priznam da sam razočaran sromaštnim *Word*-ovim filtenma za uvoz i izvoz fajlova, kao i njegovom brzinom u odnosu na *Ami Pro*. De pogledamo listu ulaznih filtera za *Ami* (ili drugim rečima, koje pakete *Ami* **direktno** uvozi – zvezdicom (*) ćemo označiti i u kojem formatu je *Ami* sposoban da ih eksportuje): *Lotus 1-2-3* (verzije 1. 1A, 2.0 i 2.01 WKS i WK1 ekstenzije), Lotus 1-2-3 (verzije 3.0, 3.1 i *Windows* verzija ekstenzije WK3), *Advance Write*, *Ami Pro* (ranije verzije), ASCII* dBASEIII dBASEIII+, dBASEIV, DCA/FFT* (Final Form Text), DCA/RFT* (Revisable Format Text) DIF, *DisplayWrite* 4 i 5, *E-Meil* Enable Versions* 1-5 do 2.5, *Executive Memo Maker* Lotus Manuscnpt* 2.0 i 2.1, MS EXCEL 3.0 i sve ranije verzije, *MS Word* 4.0 5.0 5.1 i 5.5, *MultiMate* verzija* 3.3, *MultiMate Advantage II* Navy DIF* Paradox* sve verzije do 3.5, *Peach Text* 2.11 i ranije verzije, *Rich Text Formet*, Semna Word*, *SmartWare* verzije 1. *SuperCalc* verzije 3 i 4, *Symphony* verzije 1.0 1.01 i 1.1 (ekstenzije .WRK i .WK1) *Windows Write*, *Word za Windows*, *WordPerfect* verzije 4.1, 4.2 5.0 i 5.1, *WordStar 2000* verzije 1 i 3. Neverovatan spisak, koji garantuje sigurnost u radu – više uopšte nije važno ko radi u kojem tekst procesoru (ili epliketivnom paketu). Preuzmete rad od konsnika u njegovom formatu, obradite ga u *Ami Pro*-u, i eksportujete ponovo u njegov format! Kako vam se to sviđa?

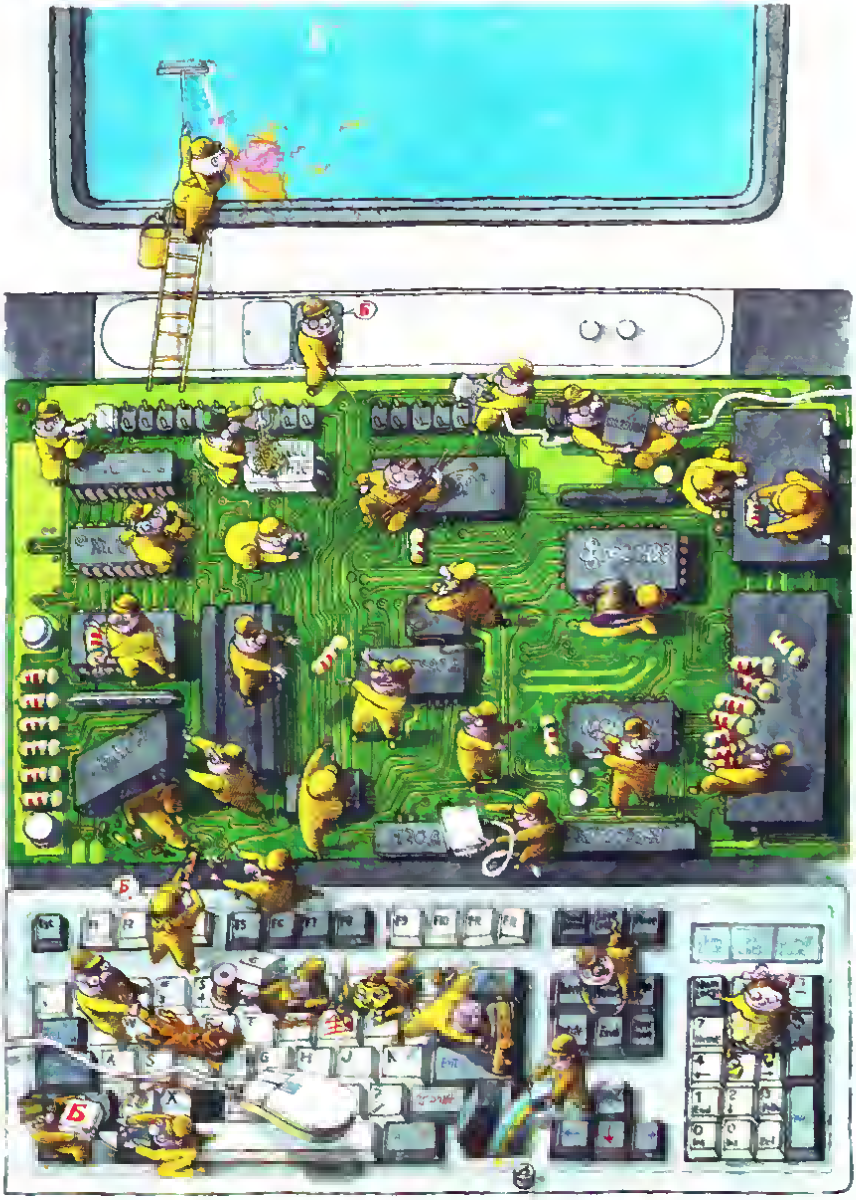
AL JE LEP OVAJ SVET

Makroi su oduvek bili posebna priča – oni dolaze kao kruna vašeg rada, dakle, podrazumeva se određen broj sati (nedelja, meseci?) rada u programu. *Ami*-jevi makroi će i ovog puta biti samo spomenuti, za nadom da će u nekom budućem prikazu dobiti dostojno mesto. Inače, automatizacija poslova nezadržljivo napreduje i samo je pitanje vremena kada će i makroi izodrti zasluženo mesto pod suncem.

Ami Pro verzije 3.0 poseduje ogromne mogućnosti. One, pre svega, leže u *Ami Pro Mecro* jeziku koji omogućava, na relativno jednostavan način, zepis mekro sekvenci (operacija koje želite da se ponavljaju u strogo utvrđenom redosledu). Mislim da se automatizacija posla ill grupe poslova kod *Ami Pro*-a približava apsolutu.

Spomenućemo još jedan segment *Ami Pro*-a koji će sigurno imati svoj poseben osvrt u nekom od naredrih brojeva „Računara", a to je rad u mreži. Po informacijama koje stižu korisnici su prezadovoljni, a i naša *Ami Pro* „kasta" je ubeđena da će *Ami Pro 3.0* u nezadržljivom jurišu osvojiti ogromno tržište.

Osećam se kao na početku. *Ami Pro 3.0* ima toliko mogucnosti, da stalno otkrivem nove i nove – grenica je vrlo daleko. Namenjen je ljudima velike kreativnosti! do maksimuma optimizovan, osloboden suvišnih poteza, odličan u performansama. Lepo je videti na knjizi natpis „Lotus Development Corporation. Word Processing Division". Lotus je dugo čekao da se uključi u trku za deo kolače na tržištu tekst procesora. All, izgleda da sad imaju Karla Luisa – čeka se odgovor „Microsofta" „WordPerfecta" i ostalih. Što se mene tiče, svaki put kad sednem za računar i počnem da radim u *Ami Pro*-u, osetim neki unutrešnji glas koji mi govori „Play it again, Sam!"



BEST
computers
011/320-103,
Beograd, Majke Jevrosime 42;

ADACOM

personalni računari, računarski inženjering, hardver, softver, instalacije, održavanje, proizvodnja i razvoj

386 SX – 25 MHz
Desktop kućište
SVGA mono
2 MB RAM
Hard disk 85 MB

1550 DEM

386 DX – 40 MHz
Desktop kućište
SVGA mono
4 MB RAM
Hard disk 85 MB

1990

PRO MUSICA

ADACOM, Čika Ljubina 12, Beograd tel./fax: 3811/629-233

IMTEL

INSTITUT ZA MIKROTALASNU TEHNIKU I ELEKTRONIKU

CAD STATION

TEL 135 420 FAX 138 928

- EISA 486-50/66, 256kB CACHE, 8-32MB
  SCSI DISK 200MB-1500MB
- TIGA VIDEO KARTICA (1400X1200)
  TMS 34020 TRUECOLOR
  MULTISYNC MONITOR 17"-20"
- EISA ETHERNET KONTROLER, MASTER
  EISA SCSI-2 KONTROLER SA 1-16MB CACHE

USLUGA ZA USLUGU

Windows for Workgroups (_WFW_) novi je proizvod „Microsoft"-e u oblasti _Windows 3.1_ aisteme. Proizvod obezbeđuje _Windows 3.1_ integralne sposobnosti za _peer-to-peer_ umreževanje, elektronsku poštu i funkcije „resporeda čaeove" za redne grupe. Namenjen je grupeme sa najviše 20 korisnika koji imeju potrebu za deljenjem informacija i reeurse. Za grupe veće od 20 korisnike, „Microsoft" preporučuje mnogo formelniji pristup, keo što je _Lan Menager_, proizvod njihove kuće. Po mišljenju predetevnike „Microeoft"-a, melim grupeme nieu potrebne ciient-server usluge.

Zoran Kohler

Pored omogućavanja _peer-to-peer_ umrežavanja, _WFW_ se izvrsno uklapa u „Microsoft"-ovu globalnu _Windows NT_ strategiju Klijentske mašine vezane na _Windows NT_ servar biće ili _LAN Manager_ klijenti, ili će imati _Windows for Workgroups_ Ono što „Microsoft" ne kaže akspliciino jeste da je _WFW_ rešenja jeftinija i jednostavnija naročito za manje zahtevne konsnike „Microsoft" namerava da ograniči broj korisnika koji se mogu priključiti na _NT_ server, prekobrojni konsnici mogu se sa ostatkom grupe povezati preko _Windows for Workgroups_

WFW se pojavio jako brzo na tržištu – oktobarski brojevi evropskih računarskoh časopisa tak su najavljivali njegovo predstavljanje na „Microsoft" ovim prezantacijama Računari" su ga dobili na testiranje početkom decembra, zahvajujući firmi „PC Club" direktno iz SAD Sve u svemu od nejave do pojave paketa proteklo je jedva nešto više od mesec dana U prodaji je kompletan paket, kao i _Add-on_ verzija koje zahteve da je na računaru ranije već instaliran standardni _Windows 3.1_ Verzije pakete sa razlikuju i po broju korisnika koji podržavaju Interesantna je verzija _WFW Starter Kit_, koja sadrži i komplet hardvera za povezivanje dva računara, dve mrežne kartice, kablove i konektore

PROLAZNE GLAVOBOLJE

U kutiji ćete naći osam 3 5" disketa, priručnika „Getting Started", „User s Guide", „Schedule+ User s Guide" i „Mail User s Guide" I uobičajena pemflate za registraciju proizvoda, zamenu 3 5" diskete od 1 44MB za diskate kapaciteta 720kB, kab i papir se odgovonma na uobičajena pitanja konsnika

Zahtevi su slični kao i ranija MS-DOS 3 3 ili noviji (5 0 se preporučuje), za _386Enhanced_ način rade potreban je 386SX procesor ili viši, sa 3MB ili više memorije dok se 4MB preporučuje Deljenja resursa moguće je samo u _386Enhanced_ modu Za one kojima deljenje resursa nija potrebno bice dovoljan I procesor 266 sa 2MB memonje Slobodnog prostora na disku mora brti dar 9 5MB iako se preporučuje bar 14 5MB Ako ja na računaru koji ima 386SX ill viši procesor _Windows 3 1_ već Instaliran treba imati bar 3MB memorija I najmanje 3 5MB slobodnog prostora na disku (preporučuje se 8 5MB) Naravno da bi računer radio u mreži mora imati i mrežnu karticu a mrežni kablovi moraju biti postavljani kako to zahteve izabram protokol i topologija mreža

Instalaciju sam obavio ne dve mašine ne prvoj je _Windows 3.1_ vec bio instaliran uz razne dodatke _Norton Desktop for Windows, Win Speed_ drajvar za SVGA kartu OEMM i drugi Drugi računar bio je prazan – samo MS DOS 5.0 U oda računara bila su installrane Ethernet mrežne kartice, kompatibilna sa poznatom „Novell" NE-2000 kartom Ni se jednim računarom nija bilo većih poteškoća pn instalaciji Prva komercijalna verzija _Windows 3 1_ nije se baš tepo slagala se _NDW_ 2 0 paketom nekon instalacije ostavljala je **Program Manager**, a na _NDW_ kao _shell_, a bilo je i slučajeva da Instalacije uopšte ne prode Sa ovom verzijom _Windows_-a takvih problema nije bilo Ono što, izgleda, zbunjuje _Windows_ pri instalaciji jeste veliki broj instaliranih fontova 155 fontova na mojam računaru **Setup** je zaboravio uz izjavu da se nalaze u WIN INI datoteci zapisani pod **[w34sfdhh4]** sekcijom I da ih moram sam kasnije dodati u **[Fonts]** sekciju **[w34sfdhh4]** sekcija sadržavala je, međutim samo jednu praznu liniju

Drugi problem javio se u vezi sa drajverom za SVGA kartu Moja kartica sa ET-4000 čipom pogoni se preko _WinSpeed_ drajvera To se nija dopalo _Windows_-u _for Workgroups_ – **Setup** je ili odbijao da startuje _Windows_ Ili je instalacija prolazila, ali nekon toga _Windows_ nije hteo da se pokrene Lek za ovo bilo je izabrati standardnu VGA rezolucju u toku instalacije koja je kasnije zamenjena rezolucijom 1024x768 uz _WinSpeed_ drajvere

Iako je „Microsoft" izbacio na tržište _Eastern European_ verzija _Windows 3 1_ samo nešto malo ranija _Windows for Workgroups_ ne poznaja opcija vezane za ove delova sveta. Jednostavno će preko već instaliranih drajvera za zemlju Jugoslaviju i srpsku tastaturu Instalirati nove drajvare zapadnog sveta Interesantno da i delje postoji program _Add 852 to DOS_ koja treba da usaglasi kodnu stranu izabranu u DOS-u i kodnu stranu koju će _Windows_ koristi za uprozorena DOS aplikacije Nejasno je zašto je u _WFW_ zadržan ovaj program koji je uveden u _Windows for Eastern Europe_.

Setup program pradiaže dva načine instaliranja **Express** i **Custom Express** opcije sa ma identifikuja hardvar i softvar vaseg računara, kontiguriše _WFW_ ostvaruja vezu vašeg računara sa mrežom i ažurira konfiguraciona datoteka Sve što morate učiniti jeste da ukucate svoje ime i odgovorita na par jednostavnih pitanja Ako slučajno nemsta dovoljno slobodnog prostora na disku, **Express Setup** će predložiti koje opciona komponenta možete izostaviti – na pnmer, modula **Screen Saver**-a bitne mape za pozadinu radna površine neka od **Accessorise** programa Iako **Express Setup** postevija vrlo malo pitanja, to ne znači da ga svaki korisnik može koristiti bez ikakvog prethodnog znanja o računeru i mreži koju instalira treba znati ima grupe portove ne koje se priključuju štampači tip mrežne kartice i njene parametra i eventualne sekundarna mreže na koje će računar biti vezivan

Zbog lošeg iskustva sa **Express** opcijom, moram pnznati da je nikada ne koristim na svom računaru naročito je na bih koristio kada sa testirani softver mora po prvi put instalirati. **Custom Setup** je, dakle opcija koju sam izabrao i za _WFW_ One je bezbednija i ako se radi o unapređenju već postojeće instalacija _Windows 3 1_ – dašavalo mi se da **Express Setup** jednostavno zaboravi sve što je ranije instalirano, uključujući tu i sve programske grupe i drajvere za hardver **Custom Setup** omogućava da se precizniia, a svakako po vašoj sopstvenoj želji podese parametri instalacije

U toku instalacije primetan je novi deo koji sa odnosi na uvođenja mrežnog rada To je podešavanje opcija mrežna kartice i tipa mreže na koju će računer biti vezan Pored vezivanja u radnu grupu _Windows_ računara, _WFW_ omogućava i vezivanje na sekundarne mreže „Novell" _NatWare_ „Microsoft" _LAN Manager_ i kompatibilne mreže („3Com" _3+Open_ „DEC" _Patchworks_ i „IBM" _LanServer_)

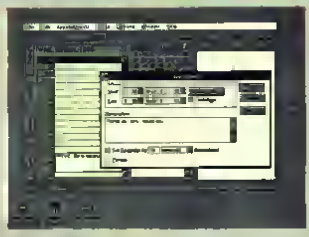
ŠTA PADA U OČI

Kako kažu u „Getting Started" priručniku, „_Windows for Workgroups_ se lako startuje" Posle uzimanje u obzir ove kratke rečenice kroz koju do nas dopire veliko iskustvo američkog pisca uputstva, otkucao sam **win** i pritisnuo **Enter** Posle uobičajenog ekrana sa logoom, pojavljuja se prvi novitet dijalog za unošenje korisničkog imena i lozinke, nakon čega

File Manager je dobio nove funkcije, posebno za rad u mreži

Program Meneger

Mail: Slanje poruke i datoteka drugim korisnicima mreže



Schedule+: Elektronski rokovnik za planiranje aktivnosti

se vaš računar prijavljuje mreži. Korisničko ime može se pokupati sa nazivom koji ste vašem računaru dali da nosi u mreži ili može biti različito – ako mašinu koristi više korisnika. Pretpostavljeno stanje je da *WFW* zahteva lozinku pri ulasku u sistem, ali se to može i eliminisati.

Nove stvari

- poodjšani **File Menager**
- poboljšani **Print Manager**
- **ClipBook Viewer** kao proširenje funkcije **Clipboard**-a
- dodatne opcije za **Control Penal**
- **Chat**, za ćaskanje na mreži
- **Net Watcher**, za pregled stanja mreže
- **WinMeter** za praćenje performansi *Windows*-a pod mrežom
- **Mail**
- **Schedule+**

Novi izgled nekih svojih delova *Windows for Workgroups* je nasledio od *Windows*-a *NT* to je novi *look* za **File Meneger** i **Print Meneger**, kao i dodatne opcije za **Control Penel**. Sa strane korisničkog interfejsa **File Mensger** i **Print Meneger** imaju sada **Button Bar** ispod menija: linija se dugmadima za aktiviranje najčešće korišćenih operacija.

File Meneger dobio je nove funkcije uglavnom vezane za rad u mreži. Ako želite da ostali članovi grupa imaju pristup nekom od direktorijuma na vašem disku, proglasićete taj direktorijum za deljeni. Deljene može podrazumevati *read-only* ili pun pristup, sa ili bez lozinke koju drugi korisnik more znati. Deljenje možete postaviti kao stalno, pa će se ono uspostavljati automatski svaki put kada uđete u *Windows*. Izbor **Connect** opcije iz **Disk** menija daje vam mogućnost da se priključite na deljene direktorijume drugih korisnika interesantno je da se na istom mrežnom kablu može naći više grupe, pa prvo birate grupu, zatim članе grupe i na kraju sam direktorijum kojem želite da pristupka. Mogućnost da slobodno roverite po disku kolege koji sedi preko puta vas sigumo će vam biti apsolutni novitet, naročito ako ste naviknuti na usluge i mogućnosti mreža tipa „Novell" *NetWare* i sličnih. **File Meneger** sada liči na istu aplikaciju u *Windows NT* sistemu, ali sa manjim mogućnostima kontrole pristupa i prava korisnika.

Print Manager nije mnogo promenjen osim što je prerušen u *NT-look*. Štampači se još uvek instaliraju i podešavaju preko **Control Penal**-a, a ne kao u *Windows NT*, iz **Print Meneger**-a. Međutim, novitete ima u vezi sa radom na mreži. Proglašavanje vašeg lokalnog printera za dostupnog svima vrlo je jednostavno i ovde se može postaviti uslov da pridošlica na vaš računar more znati lozinku, ako želi da koristi već štampač i u **Print Meneger** uveden je **Button Bar**, sa najvažnijim komandama izgled **Button Bar**-a i komande koje su na njemu može menjati iz **Optione** menija.

Koncept deležnice – **Clipboard**-a – u *WFW* je podignut na viši nivo. **Clipboard** je postao samo deo **ClipBook**-a, sveki **ClipBook** može sadržaveti više objekata koji se na njega prenose pomoću **Clipboard**-a. Svaki objekat koji se prenese na **ClipBook** naziva se stranica. Sadržaj **ClipBook**-a možete videti kao spisak sa imenima odjekata, ili ćete izabrati **thumbnell** prikaz, kada svakom objektu odgovara mala slićica sa prikazom sadržaja. Svoj lokalni **ClipBook** možete deliti sa ostalim članovima grupa, jer se **ClipBook**-u može pristupiti kad i deljenim direktonjumime **ClipBook** može sadržavati sveki valjani *Windows* objekat. Ovo otvara mogućnost da se OLE (*Object Linking and Emebedding*) koristi između dve odvojena računara. Na taj način možete objekat sa drugog računare preneti u neki od dokumenata na vašam računaru. Ako objekte povežete, sve promene koje konsnik udaljenog računara izvrši na izvornom dokumentu automatski se prenose i na sve odredišne dokumente.

Novost u **Control Panal**-u je **Network** deo, iz koga se podešavaju parametri mrežne kartice biraju dodatni tipovi mreže koje će se paralelno koristiti parametri prijavljivanja na sistem i lozinka. U **Network** opcji možete odrediti ime koje će računer nosti u grupi ime grupe, odnos vremena koje će *Windows* odvojiti za mrežne poslove i vremena za lokalne poslove, kao i da li će resursi računare biti deljeni ili ne. **Adaptera** opcija služi za menjanje parametara mrežne kartice. Ako promenite parametre *Windows* će vam ponuditi da ponovo startujete program i time preuzmete nove parametre ili nastavite sesiju sa istim parametrima. U **Logon** dalu zadajete ime pod kojim ćete biti prijavljivani na mrežu. Odavde možete dati komandu da se odjavite sa mreže. U tom slučaju korisnici sa drugih računara i dalje mogu koristiti vaše resurse, ali vi više ne moete pristupati resursima drugih računara. Izbor **Network** opcije vodi vas do dijaloge u kome možete izabrati jeden ili više tipova mreža koje će biti aktivne istovremeno kada i *WFW peer to-peer* mreža. Na spisku se nalaze *NetWare* i *LAN Manager*, a može se dodati i nako od mreža kompatibilnih sa *LAN Manager*-om.

ŽIVOT NA MREŽI

Chat je nove aplikacija za ćaskanje na mreži. Ne znam da li spade u uuslužne ili rakreacione programe – to znaju oni koji su imeli priilke da rade sa sličnim programima kao što je **Phone** pod operativnim sistemom VAX/VMS. U vremenima pre reta nije bile retkost da se započne razgovor između tri ili četiri korisnika iz cele Jugoslavije, preko JUPAK i DECNET mreža. **Chat** omogućave vazu između samo dva korisnika. Program je ukrašen uobičajenim *Windows* folklorom: ako instalirate drejver za zvučnik (nije u paketu se *WFW*) ili imate zvučnu karticu, imaćete zadovoljstvo de kada pozovete nekoga čujete zvuk koji odgovare onome što se u slušalici čuje kada dobijete broj, dok će onaj koga zovete čuti zvonjavu telefone, dosta bučnu.

Net Watcher je novi uslužni programčić (*applet* kako kaže „Microsoft") kojim možete posmetrati koje resurse – štampače, datoteke i direktorijume – konste konsnici u vašoj grupi. Naravno, ovo se dešava samo u *Enhanced* modu, jer je deljenje resursa jedino tada moguće. U prozoru se može videti lista računara koji su „zakačeni" na vaše resurse, kao i lista resursa koji su stavljeni na raspolaganje članovima grupe.

WinMeter grafički prikazuje procenat vremena procesora koje se dodeljuje procesima koji pristupaju vašim resursima kada *Windows* radi u *Enhanced* modu. Ako neko iz vaše grupe pošalje datoteku na štampanje na vašem štampaču, **WinMeter** će pokazati koji se deo procesorskog vremena angažuje dok drugi korisnik pristupa datoteci, koristi je a zatim štampa preko vašeg štampača.

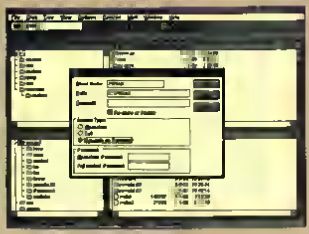
POŠTA I DNEVNI RASPORED

Mail je aplikacija za slanje poruka i datoteka drugim korisnicima mreže. Pre nego što počnete da koristite **Mail**, vi ili neko drugi iz vaše grupe morate kreirati **Workgroup Postoffice**, zajednički poštanski ured za konsnike u grupi. Onaj ko se tah tog posla (i preuzme tu odgovornost) postaje administrator **Mail**-a. Za svakog korisnika kreira se datoteka sa porukama koja postaje njegov privatni poštanski sanduć.

Kada po prvi put pokrenete **Mail**, morate se povezati sa postojećim poštanskim uredom a zatim kreirati konsnika, ako edministrator to već nije učinio. Sveki sledeći put u **Mail** ulazite tako što unosite ime i lozoniku koje ste izabrali za vaše poštensko sanduće. Ako želite, ceo postupak sa može i automatizovati za **Mail** program u **File Properties** meniju u **Commend Lina** polje na kraj linije unesite ime i lozinku, pa ćete brti pošteđeni ukucavanja svaki put pri ulasku u program. Naravno, možete izostaviti lozinku, ako dosta držite do bezbednosti vaših podataka.

Svakoj poruci može se priključiti jedna ili više datoteka. Pošto se radi o vrlo simpatičnom načinu za širenje virusa, uputstvo upozorava da te datoteke treba testreti na prisustvo virusa. U poruka se mogu ubaciti i **Embedded** objekti, što omogućava da u poštu ubacite dokumente iz drugih aplikacija, a da oni ne izgube formatiranje.

Vezivanje na mrežni printer



Deljenje prostora na disku

Ulazak u program preko lozinke

Prema dokumentaciji, radi se o aplikaciji nastalој od samostalnog „Microsoft" *Mil* paketa, koji je sada u verziji 3 0 lako perfektno funkcioniše u *WFW* okruženju, da biste se povezali na spoljne *mail* servise drugih proizvođača ili na *mail* na globalnim računarskim mrežama, paket treba unapraditi do verzije 3 0 i dodati softverski *gateway* za spoljne poštanske servise. **Mail** ne dozvoljava da jedan korisnik bude priključen na više poštanskih ureda, što u nekim situacijama može ograničiti upotrebljivost programa.

Schedule+ je elektronski rokovnik, ili raspored časova – možete ga zamisliti na oba načina, zavisno od toga za šta ga koristite. To je alat koji vam pomaža da vodite evidenciju o važnim sastancima i radnim zadacima, da izdvojite iz vaše setnice termina za sastanke, i da sami sebi ostavljate poruke o vašim dnevnim aktivnostima. Pošto *WFW* omogućava deljenja resursa, paradigma setnice može se u **Schedule+** proširiti i na resurse: vi koristite štampač od 9 do 10 ujutru, vaš kolega od 10 do podneva, itd. Pošto je prilagođen za rad u mreži, program omogućava da uvidom u setnice ostalih korisnika planirate zajedničke aktivnosti. **Schedule+** zahteva da ste ranije postali korisnik **Mail WPGO**, vaše ime za poštu i lozinku morate ukucati kada se pojavi prvi dijalog prozor.

Ekran **Schedule+** sadrži prozor setnice, se poljima za setnicu sastanaka (**Appointments**), planer (**Planner**) i listu rednih zadataka (**Tasks**). Setnica je podeljena na intervale (**slots**), od kojh svaki može da sadrži sigurni ili moguci sastanak, ili njihovu kombinaciju. Ako se neki sastanak ili obaveza ponavlja svaki dan **Schedule+** će to automatski uneti u raspored za sledeće dane. Ako ne želite da ostali korisnici vide na šta se odnosi svaka od vaših obaveza, možete ih označiti kao privatne – tada se i dalje pojavljuju u pregledu koji ostali vide, ali bez naznake aktivnosti. U raspored se unose i radni zadaci, možete im dodeliti prioritet, planirani rok završetka i grupisati ih u projekte. Kada želite da zakažete sastanak se ostalim korisnicima, to činite tako što im šaljete zahtev za sastanak, zahtev se dostavlja pomoću **Mail** programa. Pored *online* rada na mreži, **Schedule+** omogućava i *offline* rad. To ćete koristi kada radite kod kuće, ili na vaćem prenosnom računaru. Sve obaveza unesene u satnicu na ovaj način kasnije možete jednostavno priključiti satnici koju vodite na računaru u kancelariji.

RASPODELA RADNE SNAGE

Prema „Microsoft"-ovoj postavci umrežavanja, svaki računar u *WFW* grupi može istovremeno biti i server i radna stanica. Ovakav način rada jako se razlikuje od rada klasičnog servera „Novell" tipa, koji je obično gurnut u čošak sobe i na kojem niko ne radi. S druge strane, to znači da radnu snagu vašeg računara stalno delite sa korisnicima koji koriste vaše resurse. Ovo dobrom delu korisnika može predstavljati problem: još uvek se nije postiglo da standardna radna mašina bude bar PC sa 386DX procesorom na 40MHz, brzim diskom i video kartom, bar ne na našim prostorima.

Kako se *WFW* ponašao u ovom pogledu na testovima? Dobro, može se reći. Namerno sam kao računare u grupi pomešao mašine sa 386DX/40MHz i 386SX/25MHz procesorima, kao situaciju koja je unekoliko bliska realnim uslovima. Jedan računar bio je izabran za **WGPO – Workgoup Postoffice** (centralni računar za **Mail**), a na njega je bio vezan i zajednički laserski štampač. Na drugi računar u mreži bio je vezan matrični štampač, koji je imao dosta posla. Preostala dva računara nisu imala mnogo posla se zahtevima ostalih stanica. U ovakvoj organizaciji videlo se da zahtevi ostalih članova grupe ipak mogu značajano usporiti rad računara koji ih obrađuje. To se pre svega odnosi na **Mail** program, koji zbog svoje organizacije jako opterećuje procesor mašina određene da bude zajedničko poštansko sanduče. Zahtevi za štampu na laserskom štampaču nisu predstavljali veliki problem.

Treba reći da *WFW* nije za 286 računare oni mogu samo koristiti usluge drugih, a ne mogu ih sami pružati, jer je deljenje resursa moguće jedino ako procesor radi u 386 *Enhanced* modu. 386SX radi prihvatljivo, ali imajte u vidu da je za *WFW* neophodno bar 3MB slobodne memorije, ako želite da proceosr prirodno startuje u *Enhanced* modu, bez navođenja /3 opcije pri pokretanju. Uopšte, i se 386DX računarom na treba počinjati posao bez ugrađenih 4MB, a bolje je bar 8MB.

Nemojte misliti da se svi resursi mogu deliti: nisam mogao da odolim da vidim šta će se desiti ako pošaljem jedan posao za štampanje na fax/modem karticu, ugrađenu u jedan od računara. Karticu je podržavao program *WinFax* koji stvari postavlja tako da se fax vidi kao štampač priključen na jedan od serijskih portova. Na računaru se kojeg sam poslao posao nije bio instaliran *WinFax* izabrao sam štampanje preko mreže na fax karticu, poslao posao, i... ništa. Posao za štampu je jednostavno negde usput ispario. Moraću još da ispitam da li se fax karta može deliti, jer bi to predstavljalo vrlo zanimljivu stvar.

SEDI I VOZI

„Microsoft" je se *WFW* prokrijumčario i *peer-to-peer* umrežavanje pod DOS-om. Protokol drajveri, drajveri za kartice i *Workgroup* servise nisu vezani samo za *Windows*, već se mogu koristiti i iz DOS-a. **Setup** u AUTOEXEC.BAT datoteku upiše liniju **net start**, čime se zapravo iz DOS-a startuje mreža. Naredbom **net** se kasnije mogu dobiti usluge mreže vezivanje na deljene direktorijume i štampače. Sve na funkcioniše tako lepo kao pod *Windows*-om, ali je ipak tu. Doduše, na može se oprostiti potpuno izostavljanje podataka o DOS strani umrežavanja iz kompletne dokumentacije. Jedino što možete dobiti je škrti help iz samа naredbe.

Ni pojedini delovi *WFW*, naročito oni koji se odnose na rad pod mrežom, nisu dovoljno ili uopšte objašnjeni. Na više mesta u priručnicima nalaze se lakonske napomene tipa „... a o ovome možete više saznati iz **on-line help**-a." isto tako, **Setup** će u vaše CONFIG.SYS i AUTOEXEC.BAT datoteke upisati desetak linija čiji sadržaj nije objašnjen, osim što se iz naziva tih komandi može otprilike nagadati čemu služe. S ove strane, izgleda kao da je *Windows for Workgroups* na tržište izbačen u velikoj žurbi. Sa funkcionalne strane nisam imao nikakvih problema za sve vreme testiranja novog softvera.

Nakon rada se novim *Windows*-om u trajanju od par dana utisci su jako povoljni. „Microsoft" je ovim proizvodom napravio pun pogodak. *WFW* zadovoljava korienika u savremenom poslovnom okruženju, čije potrebe veće mreže prevazilaze – *NetWare*, *LAN Manager* i ostale. To se, pre svega, odnosi na situacije kada je potrebno povezati desetak PC-ja, jer se želi deljenje informacija i nekih resursa, ali ne postoji potreba za posiojanjem svih mrežnih servisa i kompletne mrežne administracije. Mislim da znate na šta mislim za pokretanje *NetWare* mreže potrebno je izdvojiti poseban računar kao server, na njemu instalirati mrežni softver, pa zatim podesiti sve parametre mreže, napraviti stablo direktorijuma na serveru, podesiti prava pristupa korienika, definisati redove za štampu, i tako unedogled. *WFW*, a druge strane, dopušta izvestan stepen prijatne anarhije: ubacite u računar mrežnu karticu, povežete računare međusobno kablovima, označite vaš štampač ili direktorijum na disku kao deljeni resurs i – počnete da radite. Sve je gotovo intuitivno, i ako imate radno iskustvo se *Windows*-om – koristi se na isti način kao i sve ostalo pod *Windows*-om. To je, naravno, i bila ideja iza celog *Windows* okruženja.

Moja preporuka, požurite i umrežite svoje računare pomoću *Windows for Workgroups*!

Korisna adresa

PC Club d.o.o.
11000 Beograd, XXI divizije 41
Tel: 011/444-30-79, Fax: 422-199

The Smart-UPS™ Series

High-performance protection for LANs, Unix systems, engineering workstations, and minicomputers

1. Smart-UPS 400
for 286, 386 desktop systems and servers

2. Smart-UPS 900
for CAD/CAM, engineering workstations and servers

3. Smart-UPS 2000
for minicomputers, multiple servers

4. Smart-UPS 1250
for multiple servers, telecom equipment

5. Smart-UPS 600
for tower-type servers and engineering workstations

Featuring

Line Interactive design
An innovative design in which the UPS inverter is connected to the output, providing a cleaner response to utility problems and superior filtering when compared to standby UPS technology

Battery replacement warning
A Smart-UPS automatically tests the health of its batteries and alerts you to potential problems before batteries wear out.

SmartBoost™
A Smart-UPS provides true brownout correction, allowing you to work through problems that shut other UPS systems down.

Sine wave output
For complete compatibility with all applications.

UPS - Link™ Control language
With a Smart-UPS, your site's power quality can actually be monitored and events can be logged automatically. You'll have hard-copy evidence of your return on investment.

Lightning, surge and noise protection
The Smart-UPS outperform other UPSs when subjected to ANSI/IEEE class "A" and "B" surge tests. Full-time EMI/RFI filters prevent line noise from corrupting data files.

Site and unit diagnostics
Automatically spot poor ground and reversed polarity, two common miswirings which usually require an electrician to diagnose.

Full approvals and warranty
UL, CSA, TUV and Novell approvals mean a safe, reliable solution to your power problems.

AUTHORIZED DEALER **"SAGA" D.O.O. BEOGRAD, YUGOSLAVIA**
Milentija Popovića 9. "SAVA CENTAR" 11070 NOVI BEOGRAD
Tel & Fax: 011/222-3579; 147-182, 222-4323 ext. 256 & 259

Plus po plus...

Računarski sistemi,

računarske komponente,

licencni software,

štampači,

najbolje cene !

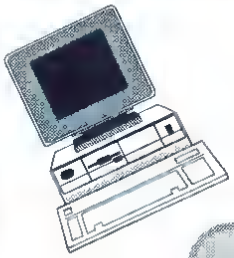
PLUS
PLUS
PLUS
PLUS

PC *plus*

Plus u Vašem džepu

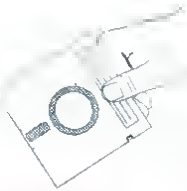
Budimska 10a, 11000 Beograd, Tel/Fax (011) 337-120, 340-227; pejdžer: 2222-302, #987

DOMINO



PONUDA MESECA

Interni streamer
IOMEGA - USA
250Mb
cena 650 DM



A CENE?

1. prelistajte RAČUNARE,
2. pronadjite najbolju ponudu
3. javite nam se!

MONITORI
HERCULES
Monohrometski VGA
Color SVGA

KARTICE...
HERCULES
VGA 256Kb
VGA 512Kb, 1Mb
VGA 1Mb, 32 hiljade boja
WINDOWS ekceleretor S3
Sound Blester
I/O+AT BUS kontroler
ETHERNET

FLOPPY, HARD, CD-ROM...
1,2Mb 5,25"
1,44Mb 3,5"
diskovi 40Mb do 1,2Gb
CD-ROM drive

OSNOVNE PLOČE
486/50, 256Kb cache
486/33, 256Kb cache
386/40, 64Kb ceche
386/33, 64Kb ceche
386SX/25
286/16

MEMORIJSKI MODULI
SIMM 4Mb/70ns
SIMM 1Mb/70ns
SIMM 256Kb/70ns

DISKETE I STRIMER-TRAKE
Diskete 1,2Mb MAXELL
Diskete 1,44Mb BASF
Diskete 1,2Mb ND NAME
Diskete 1,44Mb ND NAME
Strimer treka DC-2000, SDNY, 3M
Strimer treke DC-2120, DYSAN, CARLISLE

TASTATURE
YU etendard
ASCII standerd

RAČUNARI
286, 386, 486
sastavite konfiguraciju koja Vam odgovara!

Interni fax/modem:
2400 bauda modem
9600 bauda
send/receive fax

MS DOS 5.0
170DM

PROGRAMSKI PAKETI
DOS, XENIX, NOVELL
Programski jezici
Baze podataka
Obrada teksta
WINDOWS i aplikacije
Programi na CD ROM-u

PC CLUB d.o.o.
Beograd, XXI divizije 44
tel. 444-30-79
fax. 422-199

ISPORUKA ODMAH!

PC Club BEOGRAD
PC Club NEW YORK

ZIP JE STIGAO

Ne pamtimo da ja naki program toliko „očekivan" kao varzija 2.0 popularnog arhivera PKZIP, najavljena je još pre dve godine! Prošle godine je stigla alfa verzija, a onda smo čekali, čekali... Radovali su se jedino autori virusa koji su s vremena na vreme izbacivali „lažni" PKZIP 2.0 koji se brzo širio američkim BBS-ovima i pravio vece ili manje štete. Pred sam kraj 1992. godine firma *PKWare Inc* nas je ipak obradovala autentičnim PKZIP-om 2.04c. Jedva par dana posle svetske premijere ovaj *shareware* program je stavljen na raspolaganje i korisnicima Sezama, a verujemo da ce i čitaoci „Računara" biti zainterasovani za njegove karakteristike.

Dejan Ristanović

Istonja PKZIP-a je duga i prilično zanimljiva – pa i godina posle promocije PC računare pojavili su se prvi programi za arhiviranje podataka, a jedan od zapaženijih produkata tog tipa bio je PKARC firme *PKWare Inc* (9025 N. Deerwood Drive, Brown Deer, WI 53223, USA) – slova PK su zapravo inicijali autora programa Filea Katsa (*Phil Katz*), a ARC je trebalo da asocira na arhiviranje. Na žalost gospodina Katsa u to vreme je vec postojao arhiver zvani ARC, pa su njegovi autori tužili *PKWare* za povredu autorskih prava. Spor se otegao do 1987. godine i *PKWare* ga je konačno izgubio ne toliko zbog sličnih formale koliko zbog same ekstenzije ARC. Vansudskim poravnanjem obe strane su se složile da gospodin Kats prestane da koristi ovu ekstenziju i ovaj format iako da poslednja verzija programa PKARC (koja je kod nas i dalje u prilicnoj upotrebi) nosi datum 27. april 1987. Početkom 1989. godine *PKWare* izbacuje sasvim novi arhiver, po svim karakteristikama znatno bolji od PKARC-a, ali (zbog pomenute presude) vertikalno nekompatibilan sa njim – PKZIP. Pojavíce se nekoliko varzija tog programa da bi ona konacna 1.10 opstela na tržištu skoro tri godine – izašla je 15. aprila 1990. Za to vrame PKZIP je ušao u veoma široku upotrebu...pre svega po raznim BBS-ovima koji su ga proglasili za zvanični arhiver i rado ga naravno koriste i svi oni koji nemaju modem vec „samo" skladište vece količine podataka na diskete.

SLIČNOSTI...

Nesrevno ni konkurencija za sve to vrame nije spavala – pojavio se ARJ *Roberta Junga* koji je pokavao bržе i više od PKZIP-a i imao brojne opcije koje su povecavale komfor ali i zahtevale pamcenje raznih komandi i parametara. PKZIP je ipak „prezi veo" – dobio je poznato da je u softverskoj industriji jako vazno biti prvi. Iako da su mnogi zbog inercije nastavili da koriste PKZIP kojim su bili zadovoljni i nisu pozalili neročito keda su se pokazala neke nepouzdanosti u radu ARJ-a koje je nejzad priznao i sam njegov autor. Bilo kako bilo ljubitelji PKZIP-a su dve godine čekali novu verziju – u meduvremenu ih je *Phil Katz* obradovao samo jednom bala verzijom novog arhivera koje je nosila oznaku 1.93A. PKZIP 2.0 je najavljivan za kraj 1991. pa za početak 1992. sredinu septembar...povremeno smo na Sezam pranosili diskusije sa *PKWare*-ovog BBS-a iz kojih se moglo videti da su autori naišli na probleme sa kompatibilnošću neke bagove mikroprocesora 80486 i svašta drugo. Stpljenje je na kraju, ipak, nagrađeno – poslednji dani 1992. donesli su nam i novi PKZIP koji je verovatno da bi se razlikovao od „trojanaca" koje su razni zlonemi programeri poslednjih godina izbacivali dobro oznaku 2.04c.

PKZIP 2.04c u osnovi liči na PKZIP 1.10 pa i na stan PKARC – komandom *PKZIP ime_arhive ime_datoteke* kreiramo arhivu (standardna ekstenzije ZIP) u koju se smešte datoteke ili datoteke koje smo specificirali – *PKZIP TEST * *.*e* na primer u arhivu TEST.ZIP upisati sve programe iz tekućeg kataloge. Raspakivanje arhive obavljamo sa *PKUNZIP ime_arhive* posle čega se u tekućem katalogu pojavljuju sve zapakovane datoteke. Najzad za ZIP2EXE možemo da konvertujemo bilo koji .ZIP u izvršni program koji se po startovanju sam raspakuje – ovo je jako zgodno keda arhivu treba poslati nekome ko možda nema PKUNZIP ili ne ume da ga koristi.

Iako je ovaj osnovni oblik mnogima dovoljan PKZIP obezbeđuje razne opcije koje se navode iza crneka minus tecimo *PKZIP pr ARHIVA * *.* Samo se po sebi razume da je PKZIP 2.0 vertikalno kompatibilan sa prethodnom verzijom iako da sve poznate opcije i dalje rade. Uvedene su naravno i neke nove

... I NOVITETI

Glavni novitet je opcija **-&** koja omogućava upis arhive na veći broj disketa – često se, naime dogada da je arhivirana datoteka veća od 1.2 ili 1.44 megabajta (diskete od 360 K su već davno „zaburavljene"), što je ranije zahtevalo arhiviranje na hard disku i dalenje arhive na diskete bilo pomoću nekog specijalnog tome namenjenog programa kao što je LCOPY bilo pomoću standardnog (ili nekog komercijalnog) BACKUP-a. Ovakvo distribuiranje arhiva nije naročito pogodno pošto prmalac mora da posaduju odgovarajuci RESTORE (razne verzije DOS-a znaju da naprave ozbiljan problem) i što je ponekad još teže podrazumevali znanje potrebno da se operacija obavi. Zato novi PKZIP to radi sam – „kažete" mu *PKZIP -& A ARHIVA * *.* sve datoteke u tekucem direktorijumu bice arhivirane na diskete – računar ce tražiti nove i nove diskete dok sve što je potrebno ne bude upisano na njih.

Prirodno pitanje je šta se dogada sa onim što je već bilo upisano na te diskete. Ukoliko koristite samo opciju **-&** sadržaj disketa ostaje nedirnut – arhiva se nedovezuje na ono što vec stoji na disketi naravno u granicama slobodnog prostora. To naj češće nije ono što konsnik želi – diskete koje ce primiti arhivu bi se dakle prethodno morale obrisati što je vremenski prilično zehtevna operacija. Osim loga pocetnik ce se začuditi ako mu PKZIP zatraži mnogo više disketa nego što je predvideo – ako su diskete bile skoro pune na svaku od njih je umesto punog kapaciteta došlo svega po pai kilobajta! Zato ce se u praksi uglavnom koristiti opcija **-&w** koja briše sadržaj svake diskete (kako osnovni direktorijum tako i eventualne poddirektorijume) – komande *PKZIP pr&w A DISKC C *.** ce na primer napraviti arhivirani *backup* diska C na potrebnom broju diskete. Ukoliko su neke od tih disketa u startu bile neformatirane koristicele **-&f** umesto **-&w**

Što se rada sa više diskete liče ovo je otprilike sve što ce vam u praksi i biti potrebno. Usvojenom rešenju, ipak, nedostaje fleksibilnosi koju nudi ARJ – tamo možete da kreirate arhive proizvoljnih dimenzija keko na disketama tako i na herd disku. Ovo je jako zgodno kade, na primer, želite da pošaljete nekoj arhivu na BBS a „palite" od ogranicenog vremena na njemu – arhivu cete lako podeliti na delove čije su dimenzije usklađene sa vašim vremenskim limitom i onda pri svakom povezu sleti po jedan od njih. Ovako nesto sa PKZIP-om ne ide. Postoje i neki drugi manje vazni dodaci koje ARJ na ovom planu nudi, ali treba znati i da su ti dodaci ponešto iskomplikovali upotrebu ARJ-a. Ukratko u pitanju je vecita dilema softverske industrije – autori ARJ-a su odlučili da ponude vecu fleksibilnost uz rizik da upotrebe tih opcije postane (pre)komplikovana. Autori PKZIP-a su se odlučili za sasvim jednostavnu upotrebu ali su izgubili na fleksibilnosti i upotrebljivosti u nekim specifičnim situacijama.

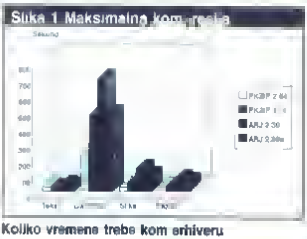
Druga zanimljiva novost je PKSFXJR – program koji omogućava kreiranje samoraspakujucih EXE datoteke koje ce po dužini biti vrlo bliske originalnom ZIP-u – pri automatskom raspakivanju lade nisu na raspolaganju sve luksuzne opcije i *help*-ovi ali sa moraspakujući EXE je najčešće namenjen nekome ko sa time lako mnogo u te opcije ne razumi. Tu je i program PKUNZIP.Jr koji zehteva znatno manje memorije od osnovnog PKUNZIP-a opet uz žrtvovanje nekih opcija. To je naročito pogodno u situaciji kada se PKUNZIP pokreće iz nekog *shell*-a pa mu nije na raspolaganju mnogo memorije.

Pomoću datoteke PKZIP.CFG arhiver se može polagoditi ukusu i potrebame korisnika – ukoliko ča sto konstite neke opcije permanentno ih aktivirajte kreiranjem ove datoteke i ubuduce štedite na kucanju! Vlesnici registrovanog PKZIP-a ovo konfigurise

	TEKST	DATOTEKA	SLIKA	RAZNO
	ASCII tekst dužine 729237	DBF datoteka dužina 9679447	TIF slika duž 1060186	87 fajlova duž 1251105
ZIP max	0:27.0 296805	4:04.0 1081071	0:28.0 39482	0:25.5 517021
2.04 at	0:19.0 295351	1:45.5 1141269	0:11.0 41857	0:26.0 520088
min	0:09.0 344618	0:55.5 1515069	0:05.5 54277	0:15.5 563399
raapak	0:05.0	0:39.0	0:05.0	0:19.5
ZIP max	0:21.0 323068	7:55.5 1368093	0:55.5 54162	0:35.5 554119
1.10 at	0:21.0 323068	7:55.5 1368093	0:55.5 54162	0:35.5 554119
min	0:07.5 361320	1:02.5 1773233	0:06.0 44588	0:14.5 638125
raapak	0:05.0	0:39.0	0:05.0	0:18.0
ARJ max	0:34.0 301757	11:37.0 1130529	2:18.5 40223	1:18.0 517901
2.30 at	0:25.0 306223	3:37.0 1224808	0:24.0 45263	0:44.5 522605
min	0:13.0 364176	1:37.5 1646040	0:09.5 59480	0:25.5 587530
raapak	0:08.0	0:45.5	0:05.0	0:24.0
ARJ max	0:27.0 301595	7:54.5 1130048	1:32.0 40240	0:56.5 517828
2.39a at	0:24.5 301707	3:22.0 1181478	0:25.5 44545	0:38.0 523877
min	0:11.0 356750	1:38.0 1646040	0:08.5 58520	0:22.0 581364
raapak	0:07.0	0:47.0	0:05.0	0:22.0

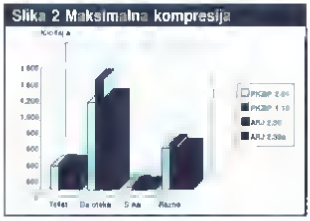
Novi šampion na svim testovime novi PKZIP pokazuje najbolje rezultate

Slika 1 Maksimalna kompresija



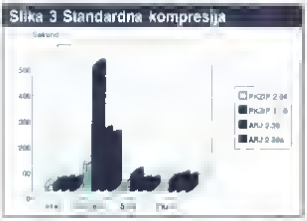
Koliko vremena treba kom arhiveru

Slika 2 Maksimalna kompresija



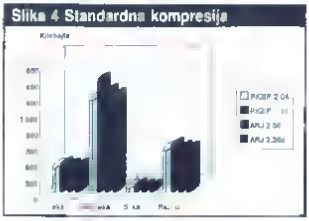
Koliko se datoteka može koprimovati

Slika 3 Standardna kompresija



Koliko vremena treba kom arhiveru

Slika 4 Standardna kompresija



Koliko se datoteka može koprimovati

nje mogu poveriti interaktivnom programu PKCFG ipak i bez njega sve može da se obavi u obicnom tekst editoru ukoliko se pažljivo pročita dokumentacija

Dosta je učinjeno i na polju sigurnosti – primenom opcija *-slozinka* datoteke mozete arhivirati tako da ih raspakuje samo onaj koma je *lozinka* poznata To je poslojalo i u ranijoj verziji ali je, prema implicitnom priznanju autora kvalitet šire bio umeran Sada se tvrdi da je za pogađanje lozinke duge samo 10 slova koja bi se sastojala od osnovnih ASCII karaktera potrebno preko 200 miliona godina Naravno kriptoanaliza nije samo probanje svih mogućih varijanti tako da u ovakve tvrdnje na treba baš uvek poverovati Ipak šifra je veoma sigurna sasvim dovoljna za amaterske primene

Drugi potencijalno možda interesantniji vid zaštite priступačan je samo registrovanim korisnicima PKZIP a Često se naime desava da se nekog manjeg BBS-e preuzmete arhivu u čiju poreklo niste bas 100% sigurni – kako da znate da je u pitanju original a ne nekakav trojanac? Ako je autor te arhive registrovao svoj PKZIP stvari bi trebale da budu znatno jednostavnije – pri registraciji navodite ime svoje firme i *PKWare* mu dodeljuje identifikacioni kod i kontrolnu sumu kojom registrujete vaš PKZIP EXE Ubuduće ce sve arhive koje kreirate sa *PKZIP I* sadržati ovaj vaš „potpis" I on će se videti pri raspakivanju – ukoliko se na kraju PKUNZIP a pojavi ime autora ili njegove firme sa prilicnom verovatnocom možete da smatrate da je arhive originalna i da niko nije „čačkao" po njoj Kažemo „sa prilicnom verovatnocom" jer je slična opcija poslojala i kod starog PKZIP e 1 10 ali je brzo „provaljena" jako da su mnogi haken ilegalno registrovali PKZIP na razna imena što je ubrzalo širenje raznih virusa i trojanaca Verovatno zbog toga novi PKUNZIP potpuno ignoriše „potpise" ugrađene stanm PKZIP-om Sada bi bar po rečima autora sve to trebalo da bude dosta pouzdanija a da li će tako stvarno biti ostaje da sačekamo i da vidimo!

Kada govorimo o novim opcijama pomenimo i nove paramatre opcije -e Pri redu se arhiverom treba naime uspostaviti ravnotežu između vremena izvršavanja i veličine finalne arhive – jasno je da se uz vise racunanja može v še kompromovati obratno PKZIP tu nudi više mogucnosti -ex je najkvalitetnija (i najsporija) kompresija -en standardna -ef brza a -es superbrza kompresija koja daje najmanji efekat

Tu je naravno i mogucnost da se sa -e0 datoteka prosto smeste u arhivu bez ikakvog sazimanja Ukoliko imate dovoljno brz računar korisćete isključivo standardnu ili eventualno maksimalnu kompresiju vlasnici XT-a ce se možda opredeliti za neki brzi i slabiji algoritam PKUNZIP automatski prepoznaje izvršenu kompresiju i raspakuje datoteka bez potrebe da se navode posebne opcije

KRALJ BRZINE

Ovako poređani novi teli PKZIP a i na izgledaju kao nešto posebno vazno Pa ipak glavna stvar nije ni u Jim korisnim dodacima ni u sigurnosti ni o ostaloj „šminki" koju smo (ili nismo) pominjali! Glavni posao arhiverа je da što pre i što više sazme datoteka A tu novi PKZIP dolazi na svoje

Kao što se vidi iz tabele I novi PKZIP je „ubio u pojam" kako svoju prethodnu verziju tako i obe „Jekuće" verzije ARJ a Što se brzine liče, jednostavno mu nema premca Kada govorimo o dimenzijama rezultujućih datoteka novi metod zvani *Deflating* nija dao neko značajno skraćenje u odnosu na ARJ ali je određena prednosti uvek (osim u jednom slučaju) bila na strani PKZIP a Kratko rečeno impresivno

Arhivere smo testirali na više tipova datoteka U prvoj koloni su podaci za ASCII tekst dužine oko 730 kilobajta (rukopis jedne knjige plus niz poruka se Sezama) u drugoj za DBF fajl dužine 9 7 megabajta (podaci o svim firmama u Vojvodini) u trecoj za sliku dužine 1 1 megabajta (tehnički crtez) i u poslednjoj rezultat kompresije sadržaja jednog „privremenog" direktorijuma našeg diska u kome je bilo 87 fajlove, od koga dosla tekstia nesto programa nesto arhiva (zip i arj) itd ukupne dužine 1 2 megabajta Sve arhivere smo testirali pri maksimalnoj podrazumevanoj i najbržoj kompresiji konsteći „standardan 386/33/64K *cache* računar sa 4 megabajta RAM-a i brzim *Conner* diskom – na 80486 se dobijaju relativno slicne proporcije ali preipostavljamo da vecina čitalaca i dalje radi na 80386 mašinama pa smo prezentirano ove rezultate

Jedina situaciju u kojoj ARJ pouzdano više kompnuje od PKZIP e pnmetio je naš kolega sa Sezama *nkbog* – arhive koje se sastoje od puno „malih" datoteka Ta usteda medutim ide na ušrb sigurnosti PKZIP ima centralni katalog koji se nalazi na kraju arhiva On mu obezbeduje lako oporavljanja arhive ukoliko je njen kraj sacuvan Ukoliko je pak centralni katalog oštećen PKZIP će poslupiti slično ARJ u i oporavljati datoteka redom U slučaju Marfijevskih oštecenja **jedne** od datoteka, sva koje su ARJ ovane iza nje biće nečitke PKZIP isti podatak ima još na jednom mestu pa je i sigurnosti veća

Toliko o kompresiju e sada da pokušavamo da odgovorimo na pitanje kako PKZIP-u 2 04c uspeva da bude **ovoliko** brzi od svih drugih programa Stvar možda i nije tako složena kao što izgleda – autori PKZIP-a odlučili su da konsle 32 bitne 80386/80486 instrukcije kao i svu memoriju (EMS XMS UMB itd) koju „nadu" u sistemu Poredtoga otkrivanje DPMI a (*DOS ProtectedMode Interface*) koji recimo podržava *Windows* garantuje prelazak u pravi 32 bitni mod uz dalje ubrzanje otkrivanje rada pod *Novell*-om može stvar dalje poboljšati Priča se da su baš ove odluke izazvale veliko kašnjenje PKZIP-a pošto kako se pokazuje nije lako radit 32-bitno pod DOS-om a isplivali su i neki *bug*-ovi samih mikroprocesora Bilo kako bilo ne verujemo da će neki buduci ARJ moći da se takmiči se PKZIP om samim poboljšanjem algoritama – moraće da „zegnze" 32-bitni „kolač"

NE MOŽE BEZ PROBLEMA

Sve ovo lepo deluje na papiru ali u praksi nije baš savršeno! Prvi i najozbiljniji problem je DMPI – pokazuje se da na mnogim sistemima u prisustvu raznih QEMM ova keš programa itd pokretanje PKZIP-a iz DOS prozora *Windows 3 1* izaziva razne smorne poruke o greškama „zaglupljuje" računar ili čak proizvodi ZIP datoteke koje se docnije ne mogu raspakovati Ukoliko se ovo dogada i na vasem sistemu ostaje vam samo da u svoj AUTOEXEC BAT ugradite *SET PKUNZIP=j* i lako nalozite programu da zanemaruje prisustvo DPMI ja

Dva korisnika Sezama i rekli b se veći broj konsnika raznih emenčkih mreza i BBS-ova prijavili su i druge probleme koji se svode na kreiranje arhiva koje se docnije ne mogu raspakovati – nije još jasno u kakvim slučajevima se to dešava i kako se može sprečiti ali izgleda da se problem najčešće javlja prilikom podele arhive na više diskeia Mi smo dosta konsili PKZIP i nismo imali slicnih iskustava ali nam je povereno je njega poljuljano ovakvim tvrdnjama – najzad nepouzdanost i jeste bila glavna mana ARJ-a 2 30! Ostaje da se svaka ZIP datoteka obavezno testira sa *PKUNZIP* – gubite malo vremena ali ste bar sigurni da nema graška!

Što se „sitnih" zamerki liče pronađena je jedna manje važna opcija ($) koja ne radi kritikovan Ireiman greška *Drive not ready* uočeno da opcija &w ne vodi racune o losim sektorima na diskeli što može da donese velike probleme Primetili smo najzad da *Norton*-ov antivirus program prijavuje da je PKZIP zaražen *Maltese Amoeba* virusom – rad se srećom o lažnoj uzbuni koja je mnoge tekako zabnula!

PKZIP 2 04c je kao i premodne verzije *shareware* program koji možete slobodno nekomercijalno koristiti i distribuirati prijateljima Komercijalna upotreba podrazumeva registraciju koja staje 47 dolara i trenutno obuhvata znatno detaljniju dokumentaciju par usluznih programa i novih opcije među kojima zapaženo meslo zauzima pomenuto -e!

Na kraju bismo rekli da nov PKZIP nesumnjivo predstavlja veliki korak unapred – radi sa o programu koji je po vecini karakteristika znatno sucenoran svim poslojećim PC arhiver ma! Videcemo kada ce i kako na ovaj izazov odgovoriti gospodin *Robert Jung* ali nema sumnje da ovo takmičenje konsnicima donosi veliku korist Ukratko pruzite šansu PKZIP u 2 04c barem do novog ARJ a!

PREUZMITE SA SEZAMA

Arhiver pkzip 2 04c nalazi se na Sezamu pod imenom pk204c.exe u direktorijumu r bmi pc\archiver* * odakle ga mogu preuzeti sv preiplatnici naseg sistema za modemske komunikacije

ПЕРИ® ХАРД ИНЖЕЊЕРИНГ

HEWLETT PACKARD Autorized Dealer

Ивана Милутиновића 24, Београд
Тел: 011/436 019, 432 319, 432 383
Факс: 011/435 513

ЗАШТИТИМО ЧОВЕКОВУ ОКОЛИНУ

ПОЈЕФТИНИТЕ РАД НА ЛАСЕРСКОМ ШТАМПАЧУ

РЕЦИКЛИРАЈТЕ ПРАЗНЕ ТОНЕР КАСЕТЕ ИЛИ НАМ ИХ ПРОДАЈТЕ

ИСПОРУКА СА СКЛАДИШТА ТОНЕРА ЗА HP-ЛАСЕРСКЕ ШТАМПАЧЕ HP-IIP; HP-IIIP; HP-II; HP-III; HP-IIID

НОВО! НОВО! НОВО!

од 1. марта рециклирамо касете за HP-LJ4

РЕЦИКЛИРАМО РИБОНЕ

ЗА ВЕЋИНУ МАТРИЧНИХ И ЛИНИЈСКИХ ШТАМПАЧА

НАЗОВИТЕ ДАНАС ЗА ИНФОРМАЦИЈЕ!

TEH NICOM

MI VAS

NEĆEMO

OSTAVITI

NA CEDILU

PANASONIC telefoni fax
EPSON printeri
HEWLWT PACKARD laser jet
BASF FUJI diskete
racunari
STAEDTLER pera za ploter
ALGOR SUPER SAP
NOVELL mreze
DATATRONICS fax modemi
usluge projektovanja
softver za knjigovodstvo

Vojvode Mišića 37
11 000 Beograd
tel (fax) 235 1108
tel 650 522/1602

software company

ProSoft

Kragujevac * JNA 63 * tel/fax: 034/212-906

"Povezivanjem Vaših računarskih ostrva možete dobiti jaku mrežu"

Direktor
Bratislav Uraković, dipl. ing

Nastavak sa strane 15

HT 340 osnovne ploča sa HT216 VGA kontrolerom 81% brža od WD 90C30 u *Windows* okruženju

BABA + ŽABA = Winmark?

Test *Winbench* daje veliku tabelu se preko 120 rezultata pojedinačnih testove Za obične smrtnike neupotrebljivo za stručnjake tvrd orah ze analizu i međusobno poređenje Testovi su vrlo raznorodni i gotovo je nemoguće sabrati ove rezultate u jedan broj Ipak u novoj verziji programa *Winbench* brzina kartice se označava sa *Winmark* i izražava u piksalima u sekundi Da bi se istakle najbitnije grafičke operacije vrši se ponderisanje rezultata Pre toga je obavljene ohim na analiza najčešće korišćenih programe da bi se odredili koeficijenti Mada ovaj pristup nije savršen i može mu se uputiti dosta primedbi za sada je jedini upotrebljiv Nerocito sa može mnogo polemisati o koeficijentima ali teško je predložiti neke druge brojeve

Nasuprot ovom pnstupu D V Veselinović svoj rezultat dobija kao aritmetičku sredinu relativnih odnosa rezultata komponente koje se testira i rezultata referentnog sistema Na taj način svi testovi dobijaju jednaku teznu u konečnom rezultatu me kolika bile ve rovatnoća njihovog pojavljivanja u programima Vrednost ovakvog rezultata ne treba komentarisati

ZAKLJUČAK

Stvar je dobrog vaspitanja da IMTEL ne hvali svoje proizvode tako da ni ja to neću učiniti ovom prilikom Neša želja nikad nije ni bila da ponudimo kupcima apsolutno najbrži ili najskuplji model već onaj koji će im za njihov novac doneti najviše performansi Na taj način treba posmatrati i ovu osnovnu ploču sa *local-bus* tehnikom To nije najbrža ploča ali svakako nije ni najskupija Postoje primene ze koje je dobra a to su prvenstveno sve GUI aplikacije

Oni koji žele da pročitaju opširniju i bolje dokumentovanu verziju ovog teksta mogu je potražiti na SEZAM-u ili IMTEL BBS-u

Literatura

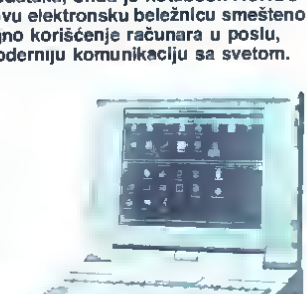
1 NEC *Memory Products Data Book*
2 Intel *i80486 Microprocessor*
3 E Solari *AT bus design IEEE P996 compatible*
4 R Ferrero *Programmer's Guide to the EGA and VGA Cards*

APP SYSTEMS
Jovana Ristića 6
11000 Beograd
tel/fax: 011 64 71 90
besplatni programi za PC računare
otkup polovne PC računarske opreme
prodaja PC računara staro za novo
prodaja repariranih PC računara
servis PC računara i opreme
diskete, riboni, papir,.....

BELEŽNICA ZA BIZNISMENE

Ako spadate u kategoriju za koju je računar alatka kao i svaka druga, a ne zbirka ovih_ili_onih tehničkih podataka, onda je notebook ACROS 325SE možda prava stvar za vas. U ovu elektronsku beležnicu smešteno je sve što je neophodno za ozbiljno korišćenje računara u poslu, svakodnevnom životu i – za najmoderniju komunikaciju sa svetom.

Ranka Jovanović



Uprkos malim dimenzijama (58×290×215 mm) i izuzetno laganom telu (6 3 funti – 2 8 kg) u ACROS je nabijene računarske snage ozbiljnih stonih modele Računar je izgrađen oko Am386SXL mikroprocesora na 25 MHz a 4 MB radne memorije (proširivo do 8) brzi tvrdi disk od 2 5 inča kapaciteta 80 MB (postoje i modeli sa 120 MB) crno-beli VGA LCD displej 640×480 tačaka se 16 svih nijansi pa čak i podnožje za matematički kopiocesor predstavljaju solidne resurse čak i za najzahtevnije operativni sistema (uključujući naravno nezaobilazni *Windows* kao i sve popularnji OS/2) i aplikacije M njelumi prostor u koji su smeštene ove komponente nimalo ne utiče na procesnu snagu računala Na uobičajenim brzinskim i aplikativnim testovima ACROS 325SE je pokazao uključujući i video podsistem rezultate u rangu (procenat manje ili više) stonih mašina 386SX klase

NEŠTO VIŠE

NiCAD batarije obezbeđuju dovoljno energije za dva i po sata rada bez priključivanja na električnu mrežu a za punjenje „do vrha" im je potrebno pola sata manje U štednji struje se vodi računa o svakom delelu – eko ključni podsistemi nisu aktivni duže od 1/16 sekunde brzina procesora se spušta na 1 64 MHz tvrdi disk zamrzave ako unosti e monitor se gasi računa mašina pada u duboki san iz koga se budi pritiskom na dugme

ACROS je opremljen svim uobičajenim interfejsima za povezivanje sa periferijskim uređajima Pored serijskog i paralelnog priključke postoji i posebna PS/2 utičnica za miša koji se tekode isporučuje u kompletu sa računarom e spada po lepoti u izuzetnije primerke svoje vrste I ne samo to Obavezan deo opreme je i internl fax/modem 9600/2400 bode čime je maš na usmerena ne poslovno Tržište Me gde se nalazili možete eko vam je pri ruci telefonski priključak poslati pismenu poruku poslovnom partneru ili se uključiti u svoju omiljenu mrežu za modernake komunikacije Priključak za štampač omogućuje da ne licu mesta odštampate ugovori il protokol o poslovnom dogovoru Uz ovakvu spravu spremi sle za poslovni komak i poslovn odnos svuda I na svakom mestu

U računare pogotovu PC klonova obično se ugrađuje samo ogoljeni DOS – tek toliko da se vidi da mešina radi ACROS je znatno bogatije opremljen softverom uz DOS na računaru se nalaze i *Windows 3 1* (koji radi veoma brzo i jako lepo izgleda ne LCD displeju) komunikac oni alat za fax/modem I *Prodigy* softver za pristup istoimenom personalnom servisu koji je namenjen potrebama poiodice i u Americi je veoma popularan M od toga na možemo imeh puno konst ali beležimo i taj podatak kao zanimljivu ideju Zašto se ubuduća ACROS ili neka druga mašina ne bi prodavao opremljen svim neophodnim alatkama i potpuno podesen za komunikaciju sa Sezamom ili – Berzom Studija B?

IZ MOG UGLA

Snaga mašine naravno nije jedina bitna komponenta za udoben rad Tastatura sa 82 tastera na kojoj nema mesta za svih desel prstiju i LCD monitor su previše mali za dugotrajan rad Pošto se mi nismo zadržali na prvim utiscima već smo na ACROS u preveli umetak o Bilu Gejtsu osetili smo u praksi ove nedostatke Zaista za onog ko inače intenzivno piše na PC računaru problem je prelazak se standardne na *notebook* tastaturu kako zbog dimenzija tako i zbog promenjenog rasporede pa i upotrebe nekih tastera (pre svega funkcijskih) U našem slučaju je bilo dosta lutanja i nehotičnih grešaka u kucanju sto je uspornlo rad i zahtevalo posebno pažljivu redakturu teksta

Prikaz je za potrebe pisanja teksta sasvim zadovoljavajući Naravno s obzirom na dimenzije i kvalitet *notebook* ekrana teže ide ispravljanje i redigovanje teksta što je u našem slučaju jedan od osnovnih poslova Kontrast je slabiji kursor je manje primetan obeleženi delovi (highlighted) su manje uočljivi u odnosu na ostalak teksta što onemogucava tzv „red-lining" odnosno unošenje korekcije „crvenom olovkom" Zato smo pribegli posebnoj raspodel posiova – kod kuće pisali prevod čuvajući se brzopletog lupanja po tastaturne (što nam na žalost nije u potpunosti uspelo) a posle na stanom računaru obavljali redaktu ru

DVA LICA

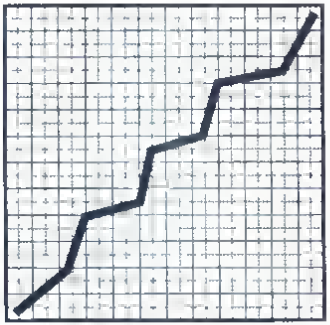
Iatini za volju ovakvi poslovi su stavili ACROS na preteranu teškoćame na beležnicu Jer *notebook* mašine i ni su namenjene za ležak fizički rad ne računaru već prvenstveno vlasnicima stonih kompjutera koji ni u jednoj prilici pa ni onde kede su u pokretu ne žele da menjaju način rada radne navika i poslovna običa je Zahvaljujuci jednostavnim dodatcima zamenom najvitalnijih delove ACROS se lako može preobrazit u stonu mašinu Računar ima priključak za spoljašnji VGA monitor i spoljašnju PS/2 (e pisko posebnog adaptera i AT) tastaturu Ovako komponovan ACROS sa ni po čemu ne razlikuje od stonog računa ra uz jednu značajnu prednost – uvek ga možete ot kačiti, staviti u tašnu čak i ze dame i poneti sa sobom na putovanje ili – poslovni sastanak

Lepi utisci pn prvom susretu s ACROS-om nisu nimalo pokvareni teškocame na koje smo naišli u prilagođavanju Te spravica koja „sve svoje se sobom nosi" pruza konsniku veliko zadovoljstvo u radu i nudi pouzdanu pratnju bukvalno svuda i na svakom mestu

Računar se prodaje po ceni od 3500 DEM

Korisna adresa

Berza Studija B
11000 Beograd Masarikova 5
Tel 333-333 lak 685 866



Promene su moguće. .
Dovoljno je samo da okrenete list!

Investirajte u BOLJE SUTRA!
Pozovite nas Rado ćemo vam pomoći.

ZNAJ MEDJU KOMPJUTERIMA
oTc COMPUTERS
MILENTIJA POPOVIĆA ◆ HAVA CENTAR ◆ 11070 NOVI BEOGRAD ◆ YUGOSLAVIA ◆ TEL 00 11 222 20 55 ◆ TELEX 12042 YU BOVCEO ◆ FAX 00 11 222 20 55 ◆ TEL 00 11 222 20 55 ◆ TEL 00 11 10 00 40 ◆ EXT 004

Saronite u svet knjiga i mikroračunara

onoliko duboko koliko želite

ABC DOS-a 5, A. Neibauer, 290 str.
ABC Lotus-a 1-2-3, C, Gilbert, 320 str.
ABC programa Quattro Pro 2, A. Simpson, 290 str.
ABC programa Word za Windows 1.1, A. Neibauer, 300 str.
ABC programa Word Perfect 5.1, A. Neibauer, 336 str.
ABC programa Windows 3.0, Kris Jamsa, 280 str.

CorelDRAW 2, S. Rimmer, 400 str.
IBM PC Uvod u rad, DOS, BASIC, S. Milinković, 400 str.
Priručnik dBASE III PLUS, D. Tanaskoski, 380 str.
CLIPPER 5 vodič za programere, R. Spence, 540 str.
Programiranje na Clipper-u, S. Straley, 780 str.

Programiranje i programski jezici

Programiranje na jeziku C, A. Hansen, 300 str.
Programski jezik C++, B. Strousturp, 300 str.
Turbo Pascal 6, S. O'Brien, 440 str.
Pascal priručnik, K. Jensen, 260 str.
Programiranje na jeziku Modula-2, N. Wirth, 190 str.
TEX za nestrpljive, P. Abrahams 350 str.
100 najkorisnijih FORTRAN-skih potprograma, L. Mendas 380 str.

Ostala izdanja

Linearna integrisana kola, B. Raković 500 str.
Elektronski hobi uređaji, D. Mijačić, 110 str.
Elektronske konstrukcije sa 555, D. Mijačić 120 str.
Arterije mozga, S. Marinković,
Commodore za sva vremena, D. Tanaskoski, 330 str.
Spektrum priručnik, V. Jankovic, 260 str.
Finalna prerada drveta, D. Skakic,

Mikro knjiga
Petra Martinovića 6
11030 Beograd
Tel.: 542 516 i 542 619

MT COMPUTERS
Preduzeće „MZ" 11000 Beograd, Dubljanska br. 70 Tel. 011/434-412
Fax. 011 450 471

ŠTAMPAČI EPSON

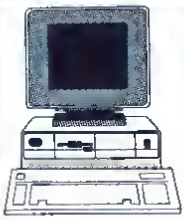
LX-400 9 pin A4 format 180 z/s ćirilat
LX-100 9-pin A4 format 240 z/s CSF YU set
LX-850 9 pin A4 format 200 z/s YU set
FX 870 9-pin A4 format 380 z/s YU set
LX-1050 9-pin A3 format 200 z/s YU set
FX 1050 9 pin A3 format 300 z/s ćirilat
LQ 100 24 pin A4 format 300 z/s CSF ćirilat ESC P 2
LQ-570 24 pin A4 format 240 z/s YU set ESC P 2
LQ-870 24 pin A4 format 330 z/s YU set ESC P 2
LQ-850 24 pin A4 format 300 z/s YU set kolor
LQ-1070 24 pin A3 format 240 z/s YU set ESC P 2
LQ 1170 24 pin A3 format 330 z/s YU-set ESC P 2
LQ 1060 24 pin A3 format 300 z/s YU set kolor
DLQ 2000 24 pin A3 format, 270 z/s YU set kolor
DFX 5000 9 pin A3 format 533 z/s YU set
DFX 8000 18 pin A3 format 1000 z/s YU set
SQ-870 48 mlaz A4 format 660 z/s YU-set INK JET
SQ-1170 48 mlaz A3 format 660 z/s YU-set INK JET
EPL 4000 LASER 300 DPI 6 s/m 1 MB H YU-set (opcija)
EPL 1300 LASER 300 DPI 6 s/m 1 MB RITech H YU-set (opcija)
EPL 8100 LASER 300 DPI 10 s/m 2 MB RIT POL 5 H YU set (opcija)
EPL 7500 POSTSCRIPT LASER MEGA2 300 DPI 6 s/m 2 MB
GT-8000 COLOR IMAGE SCANER A4 format 800 DPI

Roland DIGITAL GROUP

DXY 1100 A3 format 8 para 42 cm/s mag držanje papira
DXY-1200 A3 format 8 para 42 cm/s ele. statičko papira
DXY 1300 1MB A3 format, 8 para 42 cm s ele. statičko papira
SKETCH MATE A3 format 8 para
SKETCH MATE A4 format 8 para
DFX 2500 A2 ploter tabla 8 para 62 cm/s 1 MB
DFX 3500 A1 ploter tabla 8 para 62 cm/s 1 MB
GSX 3000 A1 „ROLL" ploter 8 para 113 cm s 1 MB aut p.v.p
GRX 300 AG A1 „ROLL" ploter 8 para 66 cm/s
DFX 4600 A0 ploter tabla 8 para 62 cm/s 1 MB
GSX 4000 A0 „ROLL" ploter 8 para 113 cm s 1 MB aut p.v.p
GRX 400 AG A0 „ROLL" ploter 8 para 66 cm/s
LTX 100 A3 format
LTX 120 A3 format
LTX 321 A1 format
LTX 420 A0 format
Data buffer SYA 550
CAMM 1 PNC 1100 PLOTER REZAČ format od 50 do 640 mm (do 1600 mm)
CAMM 1 PNC 1900 PLOTER REZAČ A0 format (do 3600 mm)
STIKA Rezac sa ugrađenim skanerom sa RS232C i programom

EIZO

EIZO 9080i 16 kolor monitor 1024x768
EIZO 6500 21" monohromatski monitor 1664x1200 (60Hz) 1280x1024
EIZO F550i 17 kolor mon. ravan ekran max 1280x1024
EIZO F750i 21" kolor mon. ravan ekran max 1280x1024
EIZO T560i 17 kolor (TRINITRON CRT) 1280x1024 CRT Trio p. 0.31
EIZO T660i 20" kolor (TRINITRON CRT) 1280x1024
EIZO GRAFIČKE KARTICE – VA30 VA41 AA51 MD B09 MD B10 MD B12



PREDUZEĆE ZA INFORMATIČKI INŽINJERING I TRGOVINU

SOFT PROJEKT

VRANESI 36215 PODUNAVCI
Tel. 036 61-847 Fax. 036 67-271

SDK VRNJAČKA BANJA
Ž.R. 61820-601-9581

1. **MODEL 5P286/16 MHz 1199**
2. **MODEL SP386SX/25 MHz 1349**
3. **MODEL 5P386/33 MHz 1799**
4. **MODEL 5P386/40 MHz 1899**
5. **MODEL SP486/33 MHz 2799**
6. **MODEL 5P486/50 MHz 3099**

Osnovne konfiguracije sadrže:
40 MB HDD, 1.2 FDD, tastaturu, herkules 14' crno beli monitor, model 286 i 386 1MB RAM-a, 386 4MB i 64KB cache, 486 4MB RAM-a i 256 cache

DOPLATE:

HARD DISK 80MB	**170**
HARD DISK 105MB	**220**
HARD DISK 120	**280**
HARD DISK 200MB	**850**
VGA MONO MONITOR	**60**
VGA COLOR MONITOR	**470**
VGA card 512KB	**80**
VGA card 1MB	**160**
Drugi FDD	**110**

Kompjuterski papir:

240×12" 1+0	**380×12" 1+0**
240×12" 1+1	**380×12" 1+1**
240×12" 1+2	

EPSON LX400/800	**520**
EPSON LX810	**570**
EPSON FX850	**1199**
EPSON FX1050	**1299**
EPSON LQ570	**1199**
EPSON LQ1050	**1799**
EPSON DFX5000	**4999**
Centroniks kabl	**25**

PANASONIC ŠTAMPAČI
LASERSKI ŠTAMPAČ HP111
LASERSKI ŠTAMPAČ HP111 P
Desk Jet HP500c

TONER ZA HP111	**230**
MOUSE GENIUS	**60**
MOUSE SEPOM	**50**
FILTER 14" MREŽA STI	**40**
FILTER 14" STAKLENI	**65**
RIBONI ZA EPSON A4	**15**
RIBONI ZA EPSON A3	**25**
DISKETE 5.25" DD	**20**
DISKETE 5.25" HD	**30**
DISKETE 3.5" DD	**25**
DISKETE 3.5" HD	**35**

UZ SVAKI RAČUNAR JEDAN BESPLATAN PROGRAM ZA KNJIGOVODSTVO PREDUZEĆA ILI RADNJE SA UPUTSTVOM ZA KORIŠĆENJE RAČUNARA.

III

CENE SU U DEM. A UPLATE SE VRŠE U DINARSKOJ PROTIVVREDNOSTI PO DNEVNOM KURSU NA DEVIZNOM TRŽIŠTU, ISPORUKA JE ODMAH PO UPLATI, GARANCIJA 12 MESECI, SERVIS OBEZBEĐEN.

PRVA LJUBAV, DRUGI PUT

Iako ništa ne može zameniti dobar kompajler, postoji još mnogo „uslužnih delatnosti" čiji se nedostatak kod pisanja fortranskih programa i te kako osecao. Rekompilacija starih kodova ili renesansa na *plusFort* način.

Saša Vlajić

Fortran veteran na sceni viših programskih jezika sa svim svojim vrlinama i manama ostavio je dubok trag kod mnogih generacija naučnika i inženjera. Veliki broj njih će reći da je on njihova prva programerska ljubav. Sadržaji hiljada i hiljada fortranskih programa i biblioteka razvijenih na nekadašnjim h-po potamskim *mainframe* računarima njihovi algoritmi i podaci predstavljaju pravo zakopano blago. Stalna potreba za više verzija jednog te istog programa kako bi se omogućilo izvršavanje na različitim sistemima PC i *mainframe* stvara još uvek prilične komplikacije.

Sagledavajući sve ovo kao i mnoge druge problame u vezi s fortranom programeri sa Salford univerziteta Velika Britanija inače specijalisti za ovaj jezik zajedno sa firmom „Polyhedron Software" odlučili su da mu obezbede dostojno mesto i na početku devedesetih novim paketom *plusFORT*. U nedovima koji slede biće dat kratak prikaz svih pakeiograma iz ovog interesantnog paketa.

SPAG – „RAZMRSIVAČ ŠPAGETI KODA"

Zadatak postavljen pred *plusFORT* je u osnovi priprama larena za naku od kompajlera koji se tiču standarda ANSI FORTRAN 77. Najveći dio posla poveran je programu SPAG (inače glavnom delu paketa. Rad se pre svega o „podmladivanju" starih programa pisanih u FORTRAN u 66. Imajući u vidu da stari izvorni kodovi (*source codes*) često deluju prilično odbojno zapeljani su i gotovo nerazumljivi kreaton su SPAG nazvali „razmrsivač špageta" (*spaghetti unscrambler*).

Pod „podmladivanjem" se podrazumeva analiziranje mnoštva isprapletanih GOTO aritmetičkih IF i ostalih naiređi karakterističnih za stari tip fortrana (slika 1a). Te formirane novog izvornog koda pomoću konstrukcija modernog strukturnog programiranja. Logički srodni delovi se grupišu a broj skokova minimizuje. Takav kod je logički potpuno ekvivalentan ulaznom „špageti kodu". Znači algoritam sadržaj ostaje nepromanjen. pa nekad čak i redosled izvršavanja naredbi. Ono što tipi izmenu je samo forma zapisa tako da je razumljivost i jednostavnost održavanja sada znatno unapređena.

Stari programi često sadrže „mrtva linija" (*dead code*) koja se nikad ne izvršavaju. SPAG je u stanju da h raspozna i po želji ukloni. Kad smo već kod programskih linija treba reći da ovaj program ima tzv. „ulepšivač koda" (*code beautifier*) sa mnoštvom opcija za kontrolu labela uvlačenja retka razmaka malog/velkog slova naredbe CONTINUE i slično. Ovo omogućuje nosiocima projekata definisanje i podržavanje internih stilova pisanja programa.

Ponekad prauređivanje originalnog izvornog koda bez ikakvog dodavanja nije dovoljno alikasno. U FORTRAN u 66 obično su programi ispadali složenji u tažnji da se višestruko iskoristi nekoliko bajtova ko da npr. da se izbegne stavljanje „N=N+1" više od jednog puta. SPAG ovo ispravlja ubacivanjem malih kodnih fragmenata na mestima gde će unapređena restrukturisanog koda biti oseino (slika 1b).

Vrlo važna osobina SPAG-a je što dozvoljava upotrebu mnogih programskih ekstenzija karakterističnih za VAX i FORTRAN 90 (DO WHILE, DO ENDDO, SELECT CASE, itd.) ta ukoliko je potrebno, konverziju u obrnutom smeru – u standardan ANSI FORTRAN 77.

Ono što vredi naglasiti je izvanredna jednostavnost upotrebe. Ulazni izvorni kod ne zahteva nikakvu posebnu pripremu. Izvršavanje SPAG-a se samo oda je to neophodno može presumeriti pomoću komandna linija ili aditovanjem ASCII konfiguracijskog fajla. Što se tiče brzina ovaj program je ispred mnogih kompajlera a očuvana je pri tom visoka pouzdanost. Deseline miliona dosad obrađenih programskih redova ve „špageti koda" to najbolje dokazuju. SPAG se nudi u kompletu sa ostalim programima pakeja *plusFORT* ili zasebno. Posloja varzije za različita platforma IBM PC DEC VAX HP Prime IBM MVS i VM Sun i druge. Svaka od njih oblikovana je tako da prihvata najčešće sve fortranske dijalekte raspoložive za upotrebu na datom hardvaru. Tako na primer IBM PC verzija razuma sve PC fortrana

QMERGE – SELEKTOR VERZIJE

Na početku je istaknuto da je pisanje više varzija jednog programa kako bi se obezbedila podrška raz-

```
13  IF|TODO.EQ.0| DO TO 12
    ACT=MOD|TODO,10|
    TODO=TODO/10
    GO TO
 |62,42,43,62,404,45,62,62,62),ACT
42  CALL COPY
    GO TO 127
43  CALL MOVE
    GO TO 144
404         N=-N
44  CALL DELETE
    GO TO 127
45  CALL PAINT
    GO TO 144
62  CALL BADACT(ACT)
    GO TO 12
127         LENGTH=LENGTH+N
144         DONE=DONE+1
    CALL RESYNC
    GO TO 13
12  RETURN
    END
```

Slika 1a. Originalni kod u FORTRANU 66 (izdvajanje decimalnih cifara iz celog broja i promanje u zavisnosti od njihove vrednosti)

```
100 IF | TODO.NE.0 | THEN
      ACT = MOD|TODO,10|
      TODO = TODO/10
      IF ( ACT.EQ.1 .OR. ACT.EQ.4 .OR.
     &     ACT.EQ.7 .OR. ACT.EQ.8 .OR.
     &     ACT.EQ.9 ) THEN
        CALL BADACT |ACT|
        GO TO 200
      ELSEIF | ACT.EQ.2 | THEN
        CALL COPY
        LENGTH = LENGTH + N
      ELSEIF | ACT.EQ.3| THEN
        CALL MOVE
      ELSEIF | ACT.EQ.5 ) THEN
        N = -N
        CALL DELETE
        LENGTH = LENGTH + N
      ELSEIF | ACT.EQ.6 ) THEN
        CALL PAINT
      ELSE
        GO TO 100
      ENDIF
      DONE = DONE + 1
      CALL RESYNC
      GO TO 100
    ENDIF
200   RETURN
    END
```

Slika 1b. Verzija koda u FORTRAN-u 77 (SPAG je umetnuo LENGTH = LENGTH + N)

ličitih platformi. do sada bilo nuzno zlo. U zavisnosti sabno. Posloje varzije za različite platforme IBM PC od stepena složenosti programa kontrola unošanja izmena je iziskivala više ili manje truda i različita pravila za dužinu reči. pristupanje sistemu davanje imena fajlovima čak i određivanje vremena i datuma nije jed nstvano za sve verzije. No ono što je najviše nervirało programera bilo je upravo saznanja da golovo ni najednostavnaja aplikacija nije univerzalna. U smislu konštenja na razl čitim sistemima pa je za n sistema potrabno kontrolisati n varijanti programa.

Rešenja ovog problema *plusFORT* nudi u v du *QMERGE*-a.

Kod pisanja aplikacija korisnik nema potrebu da ni na koj način odstup od standardnog ANSI FORTRAN-a 77 il da viši pretprocesiranje koda pre kompajliranja. U okviru izvornog koda mogu se dižati sve kodna varijanta a određeni njegovi delovi će po startanju *QMERGE* a staviljen pod znak komentara već prema tome kako se poslave logički flegovi.

Ilustrujmo ovo primarom. Na slikama 2a i 2b vidi se koja se promana dešavaju na jednom kodnom segmentu kod prebacivanja između VAX i UNIX verzija.

Sve što je potrebno kod prenosa sa jednog na drugi tip jeste startovanje *QMERGE* a i davanje adekvatna komande u komandnoj linij. Tako

```
SELECT UNIX - VAX
```

znači da se želi UNIX varijanta izvornog koda. U slučaju da ovo ne navedemo program će sam pitati

```
' Is UNIX true? (Y or N)
```

Napotrebne delove koda je umesto stavljanja pod komantari moguće potpuno odstraniti ali u tom slučaju nama više povijalka na tu varziju programa

CRAM – AUTOMATSKO DOKUMENTOVANJE

Bez statističkog prikaza odrađenih podataka danas se ne može zamisliti nijedan ozbiljniji softver. Tu značajnu dimenziju paketa *plusFORT* obezbeđuje *CRAM* program koji nakon analize naseg izvornog fortranskog koda. generiše niz izvestaja o njegovoj struktun (slika3). Oni daju pregled odozgo nadole (npr. odakle se sve poziva određeni potprogram tipa SUBROUTINE koje CALL naredbe zdaje svaki potprogram) programsko stablo (opšta struktura u obliku dijagrama) i si. Naznačavaju se rekurzivni pozivi (npr. A poziva B poziva A) i navode se rutine u rekurzivnoj palji. Identifikuju se neređena sekvence a i po bliža ukazuja na odsađena greška. Navedimo tako primer gde CALL ima različit broj argumenata od odgovarajuće naredbe SUBROUTINE – ovda CRAM eksplicitno navodi koji argument su izostavljeni.

Vrlo atraktivna je pomana CRAM-a i kao dopuna linkeru *Plink86* istog proizvođača. Uz njegovu asistenciju odnosno precizan opis kodaka za rad linkera (slika 4) sa *memframe* sistema skinuti su nek od najvećih fortranskih programa ikada primenjenih na računarima zasnovanim na mikroprocesoru 8088. Program sa 110 000 fin ja izvornog koda koji kompajliran i linkovan će EXE fajl veličina 4 MB uspešno se izvršava sa samo 512K

AUTOMATSKO REKOMPAJLIRANJE

Da li vam se ikada desilo da poste neza izmena zaboravite koja sta fajlove azurirali a koje ne? Da se takve zgode ne bi ponavljale *plusFORT* ima elikasan lak – *CMAKE* program. Jednostavnim unošenjem naredbe „C" kontrolišu se čitav direktorij sa izvornim kodovima od kojih su neki modifikovani a neki ne. *CMAKE* lad automatski iskompajlira minimalni set izvornih kodova i ažurira sve fajlove koji su s njima u vezi (kontrolišu se i INCLUDE naredbe). INCLUDE

```
DO WHILE ( todo-NE-0 )
  sct = mod(todo,10)
  todo = todo/10
  SELECT CASE (sct)
  CASE (1,4,7,8,9)
    CALL badact(sct)
    EXIT
  CASE (2)
    CALL copy
    length = lengt + n
  CASE (3)
    CALL move
  CASE (5)
    n = -n
    CALL delete
    length = length + n
  CASE (6)
    CALL prlet
  CASE DEFAULT
    CYCLE
  END SELECT
  done = done + 1
  CALL resync
ENDDO
RETURN
END
```

Slika 1c Verzije koda u FORTRAN-u 90

```
*-IF VAX
    PRINT *,'Digital VAX version 1.1
    file = '[users.john]progl.for'
*-ELSEIF UNIX,XENIX
*-  PRINT *,'UNIX version 1.1'
*-  file = '/usr/john/source/progl.f'
*-ENDIF
```

Slika 2a QMERGE getnamlist SELECT VAX

```
*-IF VAX
*-  PRINT *,'Digital VAX version 1.1'
*-  file = '[users.john]progl.for'
*-ELSEIF UNIX,XENIX
    PRINT *, UNIX version 1.1'
    file = '/usr/john/source/progl.f'
*-ENDIF
```

Slika 2b QMERGE getnamlist SELECT UNIX, VAX

lajlovi ne moraju biti u jednom direktoriju. Ima *default* kompajlera i eventualni unos opcija putem komandne linije su sva što se tiče od koristnika. Te informacije se smeštaju u kontrolni *batch* fajl pri instaliranju *CMAKE*-a. Ovaj program inače ima vlastitu bazu podataka o uzajamno povezanim fajlovima što mu obezbeđuje jednostavnost i pouzdanost.

GXCHK – JOŠ MALO KONTROLE

S ciljem otkrivanja još nekih grešaka u programu ma za čim se pokazala potreba u prvoj fazi ispitivanja *plusFORT*-a dodat mu je naknadno i *GXCHK global cross checker*. On od SPAG-a uzima tabele simbola kako bi formirao globalni pregled korištenja podataka unutar programa. Možda je to najbolje objasniti pomoću analogije sa kompajliranjem i linkovanjem. Tabela simbola su tada analogne objektnim fajlovima a *GXCHK* linkaru koji ih kombinuje u jedinstven globalni pregled.

GXCHK otkriva su između ostalog i sledeće greške

- ispuštani ili više puta definisani potprogrami
- nekorišćeni potprogrami, INCLUDE fajlovi, COMMON promenljive i COMMON blokovi
- COMMON promenljive bez dodeljene vrednosti
- COMMON ime promenljive u više nego jednom COMMON bloku
- nedoslednosti kod definisanja COMMON bloka ili liste argumenata u potprogramu
- COMMON blok ili PARAMETER definisan u više nego jednom INCLUDE fajlu

```
PWMOVE    is unresolved
  called from SCGOTO     (line   81  of
SCGOTO.FOR)
  called from SWFMAG     (line   23  of
```

Slika 3 Deo izveštaja o strukturi programa

```
BEGIN
   SECTION FILE CSDATA,SMAIN
END
BEGIN
   SECTION FILE SCGOTO,SFLABL
   SECTION FILE SGOTOA
   SECTION FILE SINDNT
   SECTION FILE SRELAB
   SECTION FILE SAKLOC
   SECTION FILE SRFIG
   SECTION FILE SRPROG,SLINK
   SECTION FILE STIDY
   SECTION FILE SWPROG
END
BEGIN
   SECTION FILE SCDEST
   SECTION FILE SCINT
   SECTION FILE SWSTUN
   BEGIN
      SECTION FILE SWINIT
      SECTION FILE SWSTAT
      BEGIN
         SECTION FILE SFRAC
         SECTION FILE SWFRAG,SPNUM
         SECTION FILE SFCHAR
         SECTION FILE SWLAAL
      END
   END
END
BEGIN
   SECTION FILE SWTOKN
END
```

Slika 4 CRAM deje upućivanja za tool linkera PLINKee

Ovaj program daje tabelu koja pokazuje koji su sve simboli upotrebljeni gde kako. Ona je od koristi kod utvrđivanja potprograma koji utiču na promenu određena COMMON promenljiva.

UMESTO ZAKLJUČKA

Navedimo na kraju zanimljiv rezultat istraživanja jednog američkog instituta. Po njemu cena razvoja i održavanja softvera u toku njegovog veka iznosi od 50 do 100$ po programskoj liniji. Uzmu li se u kalkulaciju miliona linija koje su u upotrebi jasno je zašto se neki stručnjaci žale da samo održavanje softvera progута više od 2% bruto nacionalnog dohotka u SAD. *plusFORT* hvale se njegovi kreatori što vašu investiciju. Na stranu to što više neće biti masla pri sonifikaciji sa Sizifom ili Tantalom što kod rada u fortranu donekle nije bio radak slučaj.

Lična karta

Programski paket *plusFORT – programs for programmers*

Sadržaj paketa kopije softvera na jednoj 3.5" DD disketi dokumentacija (jedna knjiga)

Cena za XT/AT PC – 0 890 (godišnje održavanje – 0 100) za 386 PC – 0 1390 (godišnje održavanje – 0 160)

Programi iz paketa se mogu nabaviti i pojedinačno a dobija se i popust u slučaju da se kupuje više primeraka. Popusi se takođe odobravaju za nekomercijalnu i obrazovnu upotrebu

Prostor na disku 1Mb

Adresa prekvodača Polyhedron Software Limited Linden House 93 High Street Standlake Witney Oxon OX 7RH U.K. tel (+44) 0865 300 579

KOMPRESIJA PODATAKA
KAKO SMRVITI SLIKU

Nova tehnika kompresije podataka, nazvana Fractal Transform, sažima grafiku u boji u odnosu 50:1, ali nije sve ni u uštedi prostora

Ukoliko nemate previše vremena i prostora na hard disku, kompresija slika se velikom rezolucijom i punom paletom boja nije nimalo jednostavna stvar. Dostupne su pravi pke za rad sa tradicionalne tehnike za kompresiju podataka koje rade sa faktorom ne sna laze se sa kompleksnošću kolor grafike baš najbolje *Images Incorporated Version 2.0* (čija je cena $295) je prvi program za kompresiju koji problem rešava pomoću metoda razlamanja.

Ovaj program baziran na *Windows*-u koristi šemu za kompresovanje nazvanu *Fractal Transform* da bi zdrobio grafičke slike u datoteke 50:1 i da bi pova čao na potpuno nov način. Razlomna kompresija postiže ove impresivne osobine tako što najpre datih kuje sadržaj, zatim sadržaj čuva u obliku matematič kih jednačina u FIF datoteci (*Fractal Image Format* koji je definisan u *Iterated Systems*) a onda iskonstruiše sliku pomoću iterativnog matematičkog modela. Program koristi softversku kompresionu mašinu nazvanu POEM koja „grize" prateći 192 MB dvo h 24 bitnih kolor fotografija i ilustracija u samo 3 MB prostora na disku!

Što se hardverskih zahteva tiče, treba imati 386 ili viši PC 4MB RAM a 3.5MB prostora na hard disku 4MB permanentne *swap* datoteke. *Microsoft Windows 3.0* ili noviji (preporučuje se 8MB RAM a 10MB HD prostora i *Windows* kompatibilni SVGA adapter).

Posto razlomna kompresija generiše veoma ma le rezolucijsk nazav sne datoteke, predstavlja relativ no novu šemu u poređenju sa JPEG (*Joint Photographic Experts Group*) standardom koji koristi pik sel bazirani DCT (*Discrete Cosine Transform*) metod da bi se oslobodio suvišnih bitova. Dodavanjem JPEG metoda kompresovanja u verziju 2.0 firma *Iterated Systems* je ovom programu otvorila put za najširu upotrebu.

Veliki proboj predstavlja *Fractal Transform Resolution Enhancement*, tehnika skaliranja koja povećava rezoluciju slike. Iz *Expand* opcije birate faktor uveća nja (2, 4, 8 ili 16) i metod (brza ili kvalitetna ekspanzija razlomljenih delova ili replikacija piksela – na žalost ove dva pojma su veoma isključiva u ovom slučaju).

U toku ekspanzije, program dodaje detalje interpolacijom vrednosti piksela, a ne jednostavnom dupli kacijom. Rezultujuća slika je oštrija od slike dobijene replikacijom piksela. Možete zatim sačuvati povećanu sliku u *Clipboard*-u ili je konvertovati u nek od podržanih formata (BMP, PCX raster, TGA ili TIFF u nekoliko dubina boja) za dalje dotenvanje u nekom programu za obradu slika, za konščenje u stonom izdavaštvu ili prezentacionoj aplikaciji.

Razlomna kompresija je asimetrični metod što znači da brže dekompresuje slika nego što ih kompresuje. JPEG je simetrični metod koji za otprilike isto vreme izvrše kompresiju i dekompresiju. Tako za aplikacije kod kojih je brzina kritična, kao što su multimedijske prezentacije, razlomna kompresija predstavlja svakako najbolje rešenje.

Prilikom testiranja ove dva metoda kompresovanja na računaru *Compaq Deskpro* 386/25 sa 8 MB RAM a dobijani su sledeći rezultati:

Bilo je potrebno gotovo 4 minute da se 1.3 MB TIFF datoteka konvertuje u 228K JPG datoteku koristeći JPEG kompresiju – dvostruko više vremena da se ista datoteka konvertuje u 148K FIF datoteku. Dakle, JPEG je brži ali *Fractal Transform* pravi manje datoteke (50:1 u poređenju sa 20:1 kod JPEG metoda). *Fractal* metod je brzi kod dekompresije – potrebno mu je 25 sekundi da prikaže FIF datoteku, dok je za prikaz JPG datoteke potrebno čitavih 2 minuta.

Glavna mana uvećavanja kod razlomnog metoda je trošenje vremena. Potrebno je 10 minuta da bi se duplirala 270K PCX slika. 51 minut da bi se učetvorostručila i gotovo 15 sati da bi se uvećala 16 puta. Ovo je delimično rešeno novim, brzim metodom i *batch* tehnikom za uvećavanje i konvertovanje slika „preko noći". Takođe, s obzirom da se kod viših rezolucija žrtvuje oštrina uvećavanja razlomnim metodom nije dealno za visoko ambiciozan rad sa grafikom. Obrađeni predmeti nisu tačkasti, ali oštre ivice postaju zamagljene.

Prednosti razlomne kompresija tako nisu baš ubedljive.

M. Jolić

RASTEGLJIVI DISKOVI

Verovatno nema korisnika računara koji ne želi da ima barem dvostruko veći disk. Možda postoji (dovoljno jeftino) rešenje i za te želje?

Dušan Popović

Stacker je programski paket koji se veoma mnogo koristi daš u našim krajevima. Pošlo do sada nismo imeli nijedan tekst o ovako eksploatisanom programu, potrudili smo se da napravimo ne samo prikaz paketa nego i kratko uputstvo za korišćenje.

Šta je *Stacker*? To je *device driver* program koji kontroliše komunikaciju između računara i diska, kompromitujući i dekompromitujući podatke u toku rada („u trku"). Rezultat njegovog rada je povećanje kapaciteta diskova (uglavnom hard-diska). Koliko će kapacitet biti povećan zavisi uglavnom od podataka koji se na disku nalaze...ali se povećanje obično kreće od 0 do 100 posto. Samim *Stackerom* ćemo se pozabaviti tek pošlo se podsetimo kakva su sve rešenja za povećanje kapaciteta diskova do sada nudena.

MALO ISTORIJE...

Hard-diskovi su do pre nekoliko godina konstili samo dve tehnologije smeštanja podataka na magnetnu ploču, pa smo ih mogli podeliti na **mfm** i na **rll** diskova. Necemo se previše bavili samim načinima kodiranja ali nam je interesantna razlika između ove dva standarda. **mfm** diskovi sadrže manje podataka na istoj površini od rll diskova, a iz toga sledi sve ostale razlike.

Prva značajnije (i primetne od strane korisnika) razlika je u brzini – **rll** diskovi (čije su informacije gušće pakovane) su po pravilu brži od **mfm** diskova. Druga razlika je u kapacitetu – očigledno da jedna ploča diska formatirana po rll standardu ima veći kapacitet od **mfm** ploče, pa sloga i ceo **rll** disk ima veći kapacitet od identičnog **mfm** formatiranog diska.

Pominjemo razne **mfm** ili **rll** formatirana diskove pa ves moze zanimali po čemu su jedan disk svrstava u jednu ili drugu grupu. Generalno odluka da li će se jedan hard-disk formatirati gušće ili ređe zavisi samo od njegovog kvaliteta – ne vredi nam da gusto „isparcelišemo" disk ako su podaci na njemu kasnije neslebilni jednom dostupni a drugi put ne. Zato se uz disk koga je proizvođač deklarisao kao **mfm** po pravilu kupuje i lakav disk-kontrola – jer da je proizvođac smatrao da on može de izdrži gušće formatle zapisa podataka, deklarisao bi ga kao **rll** i prodavao ga po većoj ceni. Za razliku od toga nije naskantno kupiti kvalitetniji disk i **mfm** kontroler ali nije ni isplativo.

Upravo je postupak koga smo spomenuli koristio mnogime da povećaju kapacitale svojih hard-diskova. kupili bi neki **mfm** disk i formatirali ga kao **rll**, a onda bi se uzdali u „stepen sigurnosti" koge je proizvodač sigurno ugradio u taj disk. Naime, **mfm** disk sigurno može da izdrži i veću gušinu zapisa od **mfm** kapacitata (i analogno toma. **rll** od **rll** a) pa je takvo formatiranje u velikom broju slučajava uspevalo. Dobijao se povecan kapacitet hard-diska ali i daleko manja sigurnost podataka na disku. Kasnije bi počali da se javljaju loši sektori, podaci bi se gubili a muke koje su imali korsnici takvog diska bi se širile kao horor priče medu ostalim korisnicima računara.

Kasnije, kada su veće firme uvidela koliko korsnici PC-računara žuda za većim kapacitetima hard-diskova, u prodaji su se pojavili razni „specijalni" kontroleri koji su na „magičan" na čin povećaveli mogućnosti diskova. Pri tome je u reklamama slejalo da je ovaj sistem „apsolutno siguran" i da sa njim nama nikakvih problema. Ukoliko sle pratili priču o mfm/rll standardima onda vam je jasno o čemu se ovde radi – opal se neko doselio da uz pomoć specijelnih kontrolere gušće pakuje podatke po disku ne poštujuci specitikacije proizvođača. Šta se ovim kontrolerima dobijalo, u odnosu na prelhodno opisen „mfm disk i rll kontroler" postupak? Pa ništa osim šlo bisle ovaj „specijalni kontroler skupo platili...

PRIHVATLJIVO REŠENJE

Očigledno je zašlo su svi pobrojani sistemi bili pogresni – svako povećanje kapacitala diskova putem hardverskih zahvata (a i povećanje gustine zapisa je hardverski zahvat) nosi sa sobom nepouzdanost, koja će pre ili kasnije rezultirati gubitkom podataka na disku. Tekođe je jesno da, kade više ne možemo da se pouzdamo u hardverska rešenja, moramo da potražimo softverska.

Stacker je program koji nam omogućava da povećamo kapacital diska čisto softverski, bez nepotrebnog napiezanja hardvera. Čim se pojavio, poslao je hit pa sada ima velki broj korisnika. Verovatno vas zanima po čemu se *Stacker* izdvaja od drugih rešenja i šta mu obezbeduje pouzdanost? Da bismo odgovorili ne to pitanje pogledaćemo kako *Stacker* radi.

Po instalaciji *Stacker* će se startovati iz **config.eye** daloteke dekla po svakom uključenju računara. On tada preuzima sve interrapta za rad sa diskom, tako da sva komunikacija izmedu diska i računara ide preko njega. Sada na scanu slupa glavni deo *Stackera*, deo za kompresiju i dekompresiju.

Verovatno sle se već sretali sa programima za kompresovanje fajlova – kao na primer PKZIP-om i ARJ-om. Oni kompresuju fajlove na nakoliko procenata prathodna valičina, a zatim su u stanju da ih nepogrešivo vrate u originalni oblik. *Stacker* takode vrši komiesiju i dekompresiju, ali u toku rada, jako da sa nama čini da su fajlovi ne disku u originalnom obliku. Time se dobija mogućnost rade sa (naizgled) većim diskom, all bez potrebe da podatka koji nam trebaju stalno arhiviramo i dearhiviramo.

Posao koji radi sam program možemo podeliti na dve slučaje: kada snimamo podetke na disk program ća ih kompresovati i snimiti ih u arhivirenom obliku, tako da zauzimaju manje mesta. Kada učitavamo podatke sa diska program će dearhivirati zapis na disku i name doslavljati raspakovane podatke, tako da mi i ne moramo znati da su bili arhivirani. Ovaj posao *Stacker* vrši do krajnjih detalja, tako da se pri animanju fajlova ažuriraju i FAT tablice direktorijumi i svi parametri fajlova. Zato on uspeva da prevari i programa koji direktno (preko BIOS poziva) pisu po disku, kao na primer COMPRESS iz *PC Tools* programa ili *Norton Disk Doktor*.

Svi podaci se snimaju u jedan fajl na duplirenom drajvu (stacvol 000) koji je stalne dužina, a sve upise u taj fajl viši sam *Stacker* brinući se o alokaciji fajla i svim ostalim „ježjskim" poslovima. Sam **stacvol** fajl se sasvim normalno upisuje u sektore diska dakle obican DOS zapis.

Sada vam je verovatno jasnija kako *Stacker* radi i zašto ga smatramo pouzdanijim od prethodnih rešenje – ne disk se smašla više podetaka tako šlo se oni kompresuju, a na tako šlo se fizički drugacija smeštaju.

KO OVO PREŽIVI...

Za razliku od mnogih drugih programa, kod *Stackera* je instalacija najkomplikovanija operacija i zahteva najviše pažnje – kada jednom uspašno instalirate *Stacker* nikakav (dobro golovo nikakav) deiju rad nije potreban. Dakle, kada jednom instalirate *Stacker* on se ponaša sasvim providno, a vama se čini da radite sa istim (samo bitno većim) drajvom. Često je instalacija i sliah od mogućih probleme dovoljan razlog za odustajanje od ovog korisnog programa. pa ćemo instalaciju malo detaljnije objasniti. U objašnjenjima koja sledе ćemo pomi njati pojam duplikacije diska, pri čemu mislimo na korišćenje *Stackera* na tom disku – ne radi se baš uvek o dupliranju (već smo rekli da procenat povećanja diska varira), ali je pojam već odomaćen, pa ga i mi koristimo.

Pre instalacije samog *Stackera* moramo se pozabaviti organizacijom diska. Verovatno znate kako se vrši particioniranja diskova (podela jednog fizickog diska ne više logičkih – C i D na primer). Dakle, morate se odlučiti kako ćete podeliti diskove i koje ćete delove dupli rati a koje ne. Naš je savet da veoma mali deo date DOS u i *boot* particiu, a ostalak *Stackeru*, lako je moguće čak i *boot* particiju dupliratl, ima puno razloga da se to ne uradi. Kao prvo, na loj particiji (C disku) je dobro smestiti DOS *Stacker* i još nekoliko programa za oporavak diskova – na primer *Norton* paket. Oni kasnije mogu dobro doći u slucaju neke greške na *Stacker* drajvu. Za C disk je dovoljno odvojiti 3 do 4 megabajta, jer se tu mogu smestiti svi pobrojani programi. Ostetak prostora se može dupliratl bez problema – znači ceo disk D ako sle diskove podelili na C i D kako smo vam savetovali. Diskovi se particioniraju pomoću FDISK programa koji je deo DOS a. Postupak samog particioniranja premašuje ciljave ovog prikaza – kako sa FDISK koristi, možete videti u nekoj od knjiga koje se bave DOS programima. Posle FDISK-a je potrebno formatirati logičke diskove – dakle **format c:** i **format d:**. Sada je još potrebno instalirati DOS i pomoćne programe...a tak zelim *Stacker*.

Instelacija samog *Stackera* počinje startovenjem INSTALL programa. Nadalje sve teče preko menija i gotovo da nema načina da pogrešnim izborom nešlo pokvarila.

Prvo pitanje je da li će u toku instalacije izbacite sve rezidentne programa ili ne. Razumljive je bojazan autora programa da bi neki rezidanlni program koji previše „pelja" sa diskom mogao da zasmeta instalaciji. Probleme bi pre svega mogli da prave razni programi

za keširanje diska. Zato ukoliko imate aktivan neki od njih – uklonite ga.

Stacker ima mogucnost da napravi veći disk od nekog praznog drajva u računaru ili da poveca neki već zauzet disk. Dupliranje već zauzetog diska vam ne bismo preporucivali – tu opciju ćete iskoristiti samo ukoliko nemate dovoljno diskete za *backup* ili ukoliko imate stvarno jakih razloga. Operaciju dupliranja praznog drajva je krajnje jednostavna i preporučljiva.

Ukoliko ipak želite da duplirate drajv na kome se nalaze fajlovi, potrudite se da na disku imate i *norton speed disk* program koji služi za defragmantaciju diskova. Ukoliko ga nemate vaši podaci će biti izgubljeni...a de vem i treba saznaćete tek po završetku kompresije. O čemu se radi? *Stacker* prvo kompresuje podeliku na disku i smešta ih u STACVOL 000 fajl na istom disku. Onda obriše deo već kompnovanih podataka a zatim nastavlja se radom dok se ceo disk ne nađe u stacvol fajlu. Takav rad garantuje veoma fragmentiran stacvol fajl pa *Stacker* pokuša da ga kompresuje upotrebom nortonovog programa za kompresiju. Ukoliko nemate taj program svi podaci će vec biti kompresovani ali ih *Stacker* neće poznavati jer je fajl previše fragmentiran. Ukratko sve miniše na probleme pa ovaj nacin dupliranja diskova izbegavajte po svaku cenu. Drugi razlog za izbegavanje ovog načina je sporost – mnogo je brze ARJ-om ceo disk skinuti ne diskala pa sadržaj po dupliranju vratiti nego dozvoliti *Stackeru* da ih kopnimuje.

Mnogo jednostavniji način je dupliranje praznog drajva – operacija traje izuzetno kratko čak i ako uzmamo u obzir vreme potrebno za vraćenje sedržaja diska iz arhive (sa diskete).

Pošto odaberete nacin kompresije morate odabrati disk na kome će se kompresija izvršiti. Pri tome nije dozvoljeno duplirati već dupliran drajv što je i logicno. Primetićete de je dozvoljeno duplirati i diskete pa možemo imati flopi od 2-3 Mb!

Zatim vas računar pita koji deo izabranog diska da buda dupliran – odgovor „ceo disk" je logican sako vas program ne sprečava da samo deo diska duplirate a ostatak konslite kao i pre instalacija.

Pre samog krairanja *stacker* diska mozete pozvati **advanced options** gde se mogu podesiti još neki parametri. Samo ime nas opominja da su ta podešavanja samo za one koji znaju šta rade i da nemе preteranog razloga za njihovo menjanje. Ipak, u naprednim opcijama ima zanimljivih mogucnosti na primer menjanje veličine klastera na disku ili manjanje podrazumevanog stepena kompresije. Vec smo rekli da *Stacker* radi tako što kompresuje podatke. Verovatno znate da se različiti podaci mogu kompresovati sa različitim uspehom. Na primer podaci koji sadrze veći broj istih delova (tekst fajlovi, na primer imaju veći broj razmaka „ ") mogu se bolje kompnimovati od **exe** programe. Podrazumevani (ili prepostavljeni) stepen kompresije je kod *Stackera* postavljen ne 2:1 što obicno ne treba menjati (zapravo na našem disku je odnos 1.7:1 ali ni odnos neznatno veci od pravog stanja ne smeta). Ukoliko na duplirani disk želite da smeštate fajlove za koje ste sigurni da se mogu bolje kompresovati (bitmap tonlovi na primer) možete povećati ovaj odnos do 8:1 ali ne treba preterivati. Ovaj odnos ne menja stvarne mogucnosti *Steckera* – on samo daje informaciju raznim **dir** i **directory maintainance** programima koliko mesta ima na disku. Ukratko mozete ovaj parameter postaviti na 4:1 ali kada disk ne bude mogao da prima više fajlova, DOS će prijaviti DISK FULL.

Pošlo ste poslavili sve parametre pozovite opciju **create stacker drive** i osteje vam samo da sačekate kraj instalacije. Koliko će instalacija potrajati zavisi uglavnom od toga da li instalirate preko praznog ili popunjenog diska o čemu smo već pričali.

U STACKER REŽIJI

Po završenoj instalaciji je potrebno reselovati računar da bi promena koje je *Stacker* uneo u **config.sys** počele da deluju. Prvo što svaki korisnik ovog programa uradi je start **dir** komande a zatim sledi divljenja zbog dobijanog prostora na disku. Već smo rekli da veličinu diska koju **dir** prijavljuje treba uzeti sa rezervom ali za početak sjajno zvuči 80 slobodnih MB na disku od 40 megabajta.

Kada se naviknete na mogućnosti novog drajva počećete da ga koristite. Rad sa dupliranim diskom se ni po čemu ne razlikuje od rada sa „normalnim" diskom i ni po cemu ne možete primatiti da je nesto menjano. Moguće su varijacije u brzini diska – što je brži računar a sporiji disk (idealna kombinacije 386sx i 28ms Seagate disk) primetićete ubrzanja u radu. To se može objasniti mogucnostima procesora da brzo (ili sporo) kompresuje podatke u odnosu na stvarnu brzinu diska. Odnosno, konšćenjem *Stackera* se rasterećuje disk a opterecuje procesor u radu sa podacima – odnos njihovih brzina odrediuje konacni ulisak. Ipak promene koje se primale nisu dramaticne – rad moze poslati malo brži ili malo sporiji ali ne drastično pa se lako možete navići.

Pri radu još nismo naši na program koji ne radi sa *Stackerom* – bilo da radi pod DOS ili *Windows* operativnim sistemom. Koliko je dobro napisan *Stacker* vidi se i po tome što on bez problema radi sa *Windows*-om. *Windows* je čitav operativni sistem pa ako uspeva njega da prevari prevarice i bilo koji drugi program.

Pošlo se duplirani disk ponaša kao običan disk normalno je da ga lako treba i održavati. Pre svega imamo problem fragmentacije diskove. On se oklanja nakim od programa za defragmantaciju (na primer COMPRESS-om iz *PC Tools* pakela) ili specijalnim SDEFRAG programom koji se dobija uz *Stacker*. Glavna razlika izmedu *sdefrag*-a i bilo kog od poznatijih defragmentalora diskova je u brzini. Naime za normalnu kompresiju *sdefrag*-u treba čitav sat pa i više pa cele ga zato verovatno koristi samo eko sle na to prisiljeni (ukoliko nemate drugi program).

U toku rada ne smete ništa raditi sa već pomenutim **stacvol 000** fajlom – on je dodu še sa *system* i *read only* atributima, ali bolje da ves i mi jos jednom opomenemo. Naime bilo kakvo menjanje (ili ne daj Boze brisanje) ovog fajla može rezultirati gubitkom podataka sa diska.

O radu sa dupliranim diskovima necemo vi še mišta reći – jednostavno, ponašaju se potpuno kao normalni diskovi.

SRAČUNATI RIZIK

Jasno je da ovakav način organizacije smeštanja podataka na disk povlači za sobom i neke rizike. Pre svega postoje veće mogucnosti grešaka, samim tim što su podaci „jace" povezeni. Ne primer na običnom disku može biti loše alociranih sektora dok na dupliranom disku *Stacker* odmah diže uzbunu i ne dozvoljave dalji rad.

Postoji više načina da dode do greške na dupliranom disku. Dovoljno je da program nešto snima na disk a nastane struje. Samim tim je kompresija stigla do neke lačke podaci su „ni snimljeni ni nesnimljeni", a FAT je nelačan. Pri sledećem ukljucanju računara *Stacker* će primetiti neloličnosti u FAT labeli i pokusati da ih popravi ali nije sigurno i da će uspeti. Ukoliko FAT nije popravljen ceo dupliranl disk će biti proglašen „samo za čitenje" (*read-only*). Kade se disk proglasi kao *read-only* to znači da su padaci sačuvani ali da nije dozvoljeno pisanje po njemu zbog njihove sigurnosti. To je veoma kritična tačka u radu sa *Stackerom* i od nje postoje dve puta ili ćete uspeti da disk vratite u normalno stanje ili ćete biti prisiljeni da radite *backup* diska ponovnu instalaciju *Stackera* pa vraćanje podalake što moze biti visecasovni rad.

Kada je već proglašen *read-only* diskom postoje načini za ispravku grašaka. Prvi put kada disk bude proglašen kao *read only* je ubedljivo najpovoljniji trenutak za pokusaj ispravljanja gresaka – svaki sledeci pot vam da je menje šanse za spas podalaka.

Postoje tri alatke koje preporučujemo za popravke dupliranih diskova. Prvi program je SCHECK, deo *Stacker* paketa. Drugi je CHKDSK deo DOS-a. Treći je NDD (*Norton disk doktor II*, deo pakata *Norton utilites 5.0*). Prvo startujete SCHECK /f. On ce vam dati objašnjenje i evenluaini savet kako da popravite greške. Na svako pitanje slobodno odgovarajte sa „yes" (kucajući celu reč a ne samo prvo slovo). Zatim (što obično saveluje i SCHECK) startujte CHKDSK /f. On ce takoda pokusati da popravi delove diska. Zatim konsultuje NDD jer on daje detaljna objašnjenja grešaka na disku. Na kraju reselujte računar da bi *Stacker* pri inicijalizaciji proverio sa kojim podacima trenutno raspolaže.

Ovaj postupak se ponavlja sve dok sva ln programa ne budu potpuno „zadovoljna" odnosno sve dok bilo kop program prijavljuje greške. Ukoliko ni posle više pokušaja ne uspete da popravite greške nemе resenja osim vec pomenulog dugog posla – *backup install* pa vraćanje na disk.

Da bisla izbegli problema sa diskom potrudite se da ne gasite računar i da ga ne reselu jete ukoliko se ne nalazila u DOS-u. Čak i ako sta u *Windows*-ima izađite u DOS pa tek tada iskljućita računar. Naime *Windows* uvek ima interno otvorene fajlove koji pri naglom gašenju mogu da poremela FAT tabelu *Stacker* drajva.

ZA ILI PROTIV?

Autor ovog teksla koristi *Stacker* vec nekoliko meseci a teksl je direklan proizvod iskustva u radu se programom. U toku konšcenje je dolazilo do raznih problema koji su uz manje ili vise truda otklonjeni. Mislimo da je dovoljna preporuka to šlo i pored sitnijih stavki „protiv" ima dovoljno razloga za delje konšćenje programa.

Ovaj program se ne koristi samo kod nas (što bi se zbog našeg standarda i cene hard-diskova moglo pomisliti) nego je veoma popularan u celom svelu. Pošlo ima ogroman broj korisnika, može se smelrati dodro istestiranim i očišcenim od bagova što je bitno kod programa koji kontroliše sve nase podatke na disku.

Lično ga smatramo dovoljno pouzdanim da (uz pogled cezniivi nogled na cene novih diskova) bude nezamenljiv. Ako ste raspoloženi da i sami isprobate ovaj program možela ga preuzeli sa Sezame (direktorijum).

PC 386SX, 386, 486 ISA, EISA, LOCAL BUS

SC3033SX	SC3040DX	SC4033DX	SC4050DX	SC4066DX2	SC L.BUS
386SX-33MHz	386DX-40MHz	486DX-33MHz	486DX-50MHz	486DX2-66MHz	486DX-50MHz
RAM 2Mb	4Mb/128Kb cache	4Mb/256Kb cache	4Mb/256Kb cache	4Mb/256Kb cache	4Mb/256Kb cache
FD 1.2 Mb	FD 1.2 Mb	FD 1.2 Mb	FD 1.2 Mb	FD 1.2 Mb	FD 1.2Mb+1.44Mb
HD 105 Mb 13 ms	HD 105 Mb 13 ms	HD 105 Mb 13 ms	HD 105 Mb 13 ms	HD 105 Mb 13 ms	HD 540 Mb 12 ms
16bit IDE CONT.	16bit IDE CONT.	16bit IDE CONT.	16bit IDE CONT.	16bit IDE CONT.	32bit 4Mb CACHE
SVGA CARD 1Mb	SVGA CARD 1Mb	SVGA CARD 1Mb	SVGA CARD 1Mb	SVGA CARD 1Mb	32bit SVGA 1Mb
14" MDND MDNIT.	14" MONO MDNIT.	14" MDND MDNIT.	14" MDNO MONIT.	14" MDND MDNIT.	15" CDLDR MDNIT.
SVGA 1024 x 768	SVGA 1024 x 758	SVGA 1024 x 768	SVGA 1024 x 768	SVGA 1024 x 768	SVGA 1280 x 1024
BABY CASE	BABY CASE	MIDI TOWER	MIDI TOWER	MIDI TOWER	MIDI TOWER
TASTATURA 101	TASTATURA 101	TASTATURA 101	TASTATURA 101	TASTATURA 101	TASTATURA 101
1.650 DEM	1.990 DEM	2.690 DEM	2.990 DEM	3.350 DEM	5.950 DEM

	CENE DELOVA	1-10 KOM.	11-30 KOM.
DISKOVI	HARD DISK 105Mb-TEAC, ALPS 13 ms	850 DEM	800 DEM
	HARD DISK 170Mb-CONNER 17 ms (serija JAGUAR)	780 DEM	830 DEM
	HARD DISK 210Mb-CONNER 12 ms	1000 DEM	900 DEM
	HARD DISK 840Mb-CONNER 12 ms IDE III SCSI	2200 DEM	2000 DEM
	HARD DISK 1.3Gb-WREN 12 ms SCSI	4200 DEM	3800 DEM
	FLOPPY 1.2Mb / FLOPPY 1.44Mb	150 / 130 DEM	130 / 120 DEM
KARTICE	16 bit 2HD, 2FD IDE CONTROLER+ 2RS232 + 1 PAR.	80 DEM	40 DEM
	32 bit 2HD, 2FD IDE CONTROLER 4Mb CACHE	350 DEM	320 DEM
	SCSI 2HD, 2FD CONTROLER 4Mb CACHE	1200 DEM	1000 DEM
	SCSI EISA 2HD, 2FD CONTROLER 4Mb CACHE	1400 DEM	1200 DEM
	ETHERNET 16 bit NE 2000 / 16 bit WD ELITE	280 / 380 DEM	280 / 350 DEM
	FAX - MODEM KARTICA MNP5	300 DEM	250 DEM
VIDEO	SVGA MONO MONITOR 14" 1024x768	300 DEM	280 DEM
	SVGA COLOR MONITOR 14" 1024x768	950 DEM	850 DEM
	SVGA COLOR MONITOR 15" 1280x1024 RAVAN EKRAN	1000 DEM	900 DEM
	SVGA COLOR MONITOR 17" 1280x1024 RAVAN EKRAN	2000 DEM	1800 DEM
	SVGA CARD TRIDENT 1Mb, 1024x768	180 DEM	140 DEM
	S3 SVGA CARD 1Mb, 1280x1024 (5x BRŽA OD TRIDENTA)	350 DEM	320 DEM
PLOČE	BOARD 386 SX-33MHz	280 DEM	240 DEM
	BOARD 386 DX-40MHz-128Kb CACHE	500 DEM	420 DEM
	BOARD 486 DX-33MHz-256Kb CACHE	1200 DEM	1000 DEM
	BOARD 486 DX-50MHz-256Kb CACHE	1600 DEM	1400 DEM
	LOCAL BUS 465 DX-50MHz-256Kb CACHE, 3x32bit SLOT	1900 DEM	1700 DEM
	BOARD 486 DX2-65MHz-256Kb CACHE	1900 DEM	1700 DEM
PRINTERI	HP IIIP LASER PRINTER 300x300	2900 DEM	2800 DEM
	HP IV LASER PRINTER 600x600	3900 DEM	3600 DEM
	EPSON PRINTER LX 400 A4, 9 PIN	500 DEM	450 DEM
	EPSON PRINTER LD 570 A4, 24 PIN	900 DEM	850 DEM
	EPSON PRINTER LD 1070 A3, 24 PIN	1300 DEM	1250 DEM
	EPSON PRINTER FX 1050 A3, 24 PIN	1250 DEM	1150 DEM
RAZNO	MEMORIJE SIMM 1Mb / SIMM 4Mb	75 / 300 DEM	65 / 250 DEM
	MAT. KOPRDCESDRI IIT 387SX-33MHz / IIT 387DX-40MHz	180 / 200 DEM	180 / 180 DEM
	BABY CASE + 200W / MIDI TOWER + 200W	160 / 220 DEM	140 / 200 DEM
	PROFESIONALNA TASTATURA 101	70 DEM	60 DEM
	MIŠ - MICROSOFT KOMPATIBILAN	50 DEM	40 DEM
	STREAMER 120Mb / 250Mb	600 / 800 DEM	550 / 750 DEM

ISPORUKA ODMAH **GARANCIJA 12 MESECI**

ŽIRO RAČUN: 60815-601-64787 | RADNO VREME 9h-17h

TEL: 011/332-607
FAX: 011/345-126

PREDUZEĆE
ŠUTLIĆ
&
MIKRO DIZAJN

11000 BEOGRAD
Kosovska 32, I sprat

FEBRUAR 1993.

MR Systems & ISC

SISTEM INŽENJERING INFORMACIONIH SISTEMA
SERVERI•CAD STANICE•RADNE STANICE•TERMINAL SERVERI
TERMINALI•SCO UNIX•TCP/IP•NOVELL LITE•UPRAVLJANJE
POSLOVANJEM•PROJEKTOVANJE PROIZVODA I TEHNOLOGIJA
UPRAVLJANJE TEHNOLOŠKIH PROCESA

386SX 386DX 486DX ISA - LOCAL BUS - EISA

IZVOD IZ NAŠEG CENOVNIKA RAČUNARSKIH DELOVA

PRINTER HYUNDAI A3 9P	440 DEM
PRINTER HYUNDAI A4 9P	330 DEM
LAN KUĆIŠTE 60W	200 DEM
SLIM LINE KUĆIŠTE 200W	200 DEM
DESKTOP KUĆIŠTE 200W	145 DEM
MINI TOWER KUĆIŠTE 200W	155 DEM
TOWER KUĆIŠTE 230W	275 DEM
BIG TOWER KUĆIŠTE 300W	880 DEM
SVGA CARD TRIDENT 512K	120 DEM
SVGA CARD TSENG 1Mb	190 DEM
MONO VGA MONITOR	275 DEM
SVGA MONITOR 1024X768	590 DEM
INTERNI MODEM MNP5	150 DEM
INTERNI FAX MODEM MNP5	255 DEM
EXTERNI MODEM MNP5	215 DEM
EXTERNI FAX MODEM MNP5	330 DEM
POCKET FAX MODEM MNP5	330 DEM

POPUST ZA DILERE DO 10%

KOMPLET SVGA Monitor, Mini Tower Kućište, YU Tastatura **750 DEM**

Lomina 36 tel. 659-506 505-785 fax. 687-285

ERC commerc

MEŠOVITO PREDUZEĆE EXPORT-IMPORT

KOLARČEVA 4/IV
BEOGRAD
TEL 011 632 532
011 626 792
FAX 011 633 059
ŽIRO RAČUN
60803-601-118032

RAČUNARI

— Kompjuter 286/20 MHz	1 290 DEM
1MB RAM, Hard disk 40 MB, Flopi disk 5,25" 1 2 MB, VGA Mono	
— Kompjuter 386SX/33 MHz	1.800 DEM
— Kompjuter 386DX/40 MHz sa 128 Kb Cache	2 000 DEM
— Kompjuter 486DX/33 MHz sa 256 Kb Cache	2.890 DEM
— Kompjuter 486DX/50 MHz sa 256 Kb Cache	3 390 DEM
— Kompjuter 486DX2/66 MHz sa 256 Kb Cache	3.640 DEM
4MB RAM, Hard disk 40MB, Flopi disk 5,25" 1 2MB, VGA Mono	
DOPLATA za SVGA Color monitor 14" PHILIPS BRILLIANCE i grafičku karticu TRIDENT 512 Kb rezolucije 1024*768	490 DEM

KUĆIŠTA

BABY Flip Top	150 DEM
MINI Tower	180 DEM
SLIM Line	200 DEM
LAN Stanica	210 DEM
BIG Tower	330 DEM
SERVER BIG Tower	450 DEM

MONITORI

VGA Mono DTK	280 DEM
VGA Mono PHILIPS	400 DEM
SVGA Color ROYAL 14"	650 DEM
SVGA Color PHILIPS 14"	800 DEM
SVGA Color PHILIPS 14" LOW Emission	1 080 DEM

MEMORIJA

SIMM 1MB 70ns	80 DEM
SIMM 256*4 70ns	95 DEM
SIMM 4MB 70ns	280 DEM
DRAM 256*8 70ns	85 DEM

KONTROLERI

IDE BUS 2s/1p/1g	55 DEM
IDE BUS ISA 1MB Cache	850 DEM
SCSI-2 ADAPTEC 1542B	590 DEM
SCSI EISA 1MB Cache	1.300 DEM

MAINBOARD

80286/20 MHz	180 DEM
80386SX/33 MHz	350 DEM
80386DX/40 MHz 128 Kb	550 DEM
80486DX/33 MHz 256 Kb	1 350 DEM
80486DX/33 MHz EISA	2.200 DEM
80486DX/50MHz 256 Kb	2 050 DEM

HARD DISKOVI

42MB CONNER	390 DEM
85MB WD	540 DEM
125MB WD	790 DEM
212MB WD	1 200 DEM
520MB FUJITSU	2.600 DEM
340MB SCSI MAXTOR	2 500 DEM

PRINTERI

STAR LC-20, A4	550 DEM
PANASONIC 1123, A4	800 DEM
EPSON LQ 1070, A3	1.450 DEM

VIDEO KARTA

TRIDENT VGA 256Kb	95 DEM
TRIDENT SVGA 512Kb	130 DEM
TRIDENT SVGA 1MB	210 DEM

OSTALA OPREMA

LASER HP IIIp	2.900 DEM	Flopi disk 5,25" ili 3,5"	150 DEM
TASTATURA CHICONY	88 DEM	EPROM PROGRAMATOR	390 DEM
I/O KARTA 2s/1p/1g	40 DEM	CENTRONIKS KABL	20 DEM
NOVELL KARTA NE2000	455 DEM	MOUSE SA PADOM	80 DEM
NOVELL KARTA NE1000	425 DEM	DISKETE FUJI 5,25" 2HD	2.3 DEM
DISKET BOX 70 kom 5,25"	65 DEM	NAPAJANJE 200 W	80 DEM
DISKET BOX 150 kom 3,5"	65 DEM	NOSAC ZA HARD DISK	6 DEM

ERC POPUŠTA U SVEMU — OSIM U KVALITETU

LEGO JEZIK

U svetu postoji nekoliko stotina programskih jezika, svaki za svoju namenu, ali Srdan Mijanović, koga se stariji čitaoci svakako sećaju po „multiprocesorskom spektrumu", smatra da još uvek ima dovoljno prostora i za inovacije u ovoj oblasti. Donosimo prikaz njegovog jezika *Manevro*, čija je osnovna osobina mogućnost nadogradnje do onog oblika koji korisniku najviše odgovara. Jezik će se uskoro naći i na Sezamu, i tada će svi koje ova tema zanima moći čitavu stvar da ispitaju i uživo.

U oblasti sistemskog programiranja dugo se koristilo samo asembler ali se posle uvođenja C jezika situacija malo popravila. Međutim ne C u je moguće pisati OS ali ne i BIOS. Znači ukoliko sta suočen sa problemom kreiranja kompletnog računara popularnosti C a vam nece pomoć da izbegnete asembler.

Sa druge strane postoje mnoge rutine koje ljudi pišu za različite računare na više mašinskih jezika, a ona u suštini obavljaju iste stvari. Prevod se jednog mašinca na drugi obično polaze duze nego pisanje novog programa. Što je opet beskorisno.

To su dva glavna razloga nastanka ovog programskog jezika. Predviđen je za rad sa malo memorije (danas da ispišete ZDRAVO na ekranu najšeđi ljivi kompajler uzme 5KB?!) a veoma lako se prilagođava da proizvodi kod za druge procesore. Isto tako sintaksa mu je promenljiva i to ja jedan od onih programa koji pišu sami sebe. Ij dodaju nova instrukcija. Konsti RPN notaciju (veoma nepopularno među širokim programerskim masama) nudi mogućnosti diraklnog uključivanja asemblerskog programa dovoljno je prijav za tad a iznad svega je klasno bazirani sistem. Što omogućava naknadno definisanje stringova FP brojeva pointera i svega ostalog. kao da su sve te strukture oduvek bile tu.

RPN NOTACIJA

Konsnici paketa koji su bliski sa logikom RPN (Reverse Polish Notation – obrnuta poljska notacija) ili sa HP kalkulatorima (koj po pravilu konsta ova) oblik zapisivanje izraza) mogu preskočiti ovo poglavlje. Za one ostale racimo da bilo koji izraz možemo napisati na tri načina prefiksnom infiksnom i postfiksnom notacijom operacija.

Razmotrimo primer matematičke formule

(A+B)/(C+D) (1)

Ukoliko se opredelimo za prefiksnu notaciju i pisali postavimo da su + i / funkcije dva argumente isti oval izraz možemo zapisati u obliku

/((+ A B) (+ C D)) (2)

Kao što se vidi svaka operacija ima svoje argumente koji se navoda posle identifikatora funkcije za to ovakvu reprezentaciju nazivamo PREFIKSNA (jer operacija da PRE argumenata).

Izraz (1) je neplsan u infiksnoj notaciji. Vidi se da su + i / binarni operatori a njihovi argumenti se pišu sa njihova leve i desne strane. Ovakav način zapisa je najbliž našem načinu razmišljanja. ali se on u toku kompajliranja ili interpretiranja obavezno pretvara ili u prefiksni il u postfiksni oblik koji je lakši za mašinsku manipulaciju.

Kao što je i logično postfiksn način zapisivanja lde trećim mogućim putem argumenti se navode pre same operacije. Izraz (1) bi sada glasio

A B + C D + / (3)

Manevro (i ostali jezici koji konsle postfiksnu notaciju) je dizajniran tako da se izrazi izvorno pišu u postfiksnoj notaciji. Takav način zamaia programiranju logiku lokom programiranja (dok se ne navikne) ali značajno ubrzava izvršavanja. Postoji još jedan mnogo značajniji razlog zasto se konsti RPN (što je sinonim za postfiksnu notaciju) uvođenje novih operacija. U principu. svejedno je da li je + (iz primera (3)) matemat čka operacija ili ona ima neka sporadne efekte ili možda i nema veze sa sabiranjem. pisanje novanja operatore u većini programskih jezika koj ko nste infiksnu notaciju nije moguće.

Tima sa programaru daja sloboda da preimenuje osnovna operacije da ih unapredi (tako recimo nisu izvedene sa dovoljnom tačnošću) i sl.

Srđan Mijanović

Konkretno *Manevro* mora konstiti RPN zbog mogućnosti menjanja same sintakse jezika (kao što ćemo videti kasnije). bilo kakav proces preiprocesiranja (uključujuć pretvaranje iz niiksne u postfiksnu notaciju) postaje nemoguć

Pojam koji je narazdvojan od postfiksne notacije jesle stek. Stek je struktura podataka gde princip podataka ide po sistemu zadnji unutra-prvi-napolje (LIFO – *Last In First Out*). Sam DOS (ili bolje reći BIOS) vašeg računara podržava tzv. sistemski stek na njemu se drže neki privremeni podaci i povratne adrese prilkom poziva potprograma. U principu ovaj stek nam bi trebalo dinali bilo kakvi zahvati na njemu mogu dovesti do kraha sistema. Primetimo da su naše operacija + i / iz primera (3) kao stvorene za rad sa stekom zamislimo da je operacija + izvedana tako da skine dva broja sa steka sabere ih i rezultat stavi opet na stek. Vidimo da je rezultat sekvence ista kao i u primeru (1).

Postoji algoritam koji ce svaki izraz iz niiksne notacije prevesti u postfiksnu (i obrnuto) u stvari prilkom izračunavanja niiksne notacije računar ja prvo pretvori u postfiksnu koje je bliža njegovom shvatanju izraze

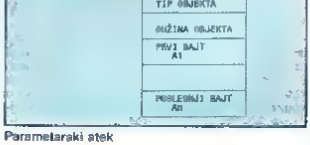
Manevro sistem konsti za mleme potreba dva steka koji se nazivaju povratni stek i parametari stek (*Return Stack* & *Parameter Stack*) Povratni stek ima ulu funkciju kao I sistamski stek na njemu se nalaza povratne adrese za izlaz iz potprograma kao I brojač **do_loop** petlje. U pnne-pu operacije se ovim stekom nisu dozvoljene (ili bar nisu preporučljive)

Parametar stek je ostavljen programeru prenosi svih parametara između operacija se odvijaju preko ovog steka. Tu je i nakolko osnovnih operacija za rad sa njim. Inače. *Manevro* na ovom staku može držati bilo koju strukturu podataka koja je dugačka do 255 bajtove normalno takvu strukturu je poliebno i opisati. Što ćemo videti kako se radi malo kasnije. Za sada recimo da se podac ma na steku može manjati redosled i da se mogu duplicirati. Tim operacijama mogu će je na staku napraviti bilo koji željeni razmeštaj podataka.

STEK

Manevro na svom parametar steku kao što smo već napomenuli može držati bilo koju strukturu veličina 1 255 bajtova (objekti veličina 0 bajtova nisu dozvoljeni) Njihove reprezentacija na steku je kao na slici 1

Slika 1



Parametarski stek

Tip objekta je informacija za sistem koja obezbeduje hilozofiju klasno bazirangog sistema. Jedina klasa sa kojom *Manevro* zna da radi u početku je tzv. klasa *integer* (celi brojevi u rasponu od 32767 do +32768). Međutim ovaj sistem je otvoren ka proširenjima pa je moguce definisati recimo stringova (koja kao strukturu podataka poznaje gotovo svaki jezik) tako da ih kompajler prepoznaje kao da su oduvek bili tu!

Programeri nizovima akcija dodeljuje imane koja zovemo reči. Postoji pet osnovnih reci za manipulisanje sa vrhom parametar steka u *Manevro* programskom jeziku *(dup drop swap rot over)* Praiko ovih pet reči je moguće vršiti man pulacije sa prva in objekta na staku. Pošto većine reči radi se dva ili jednim objektom (to su ekvivalentni unarn h i binarn h operacija) odigledno je da su ove operacije sasvim dovoljne. Međutim lako je stek preporučeno mesto za držanje objekata sigurno nije i jedino (jer šta bi se desilo da na steku mamo više od ln objekta?) Zbog toga su uvađena dva vrsta memorijskih referenci konstante i varijabla Logično konstante i varijable mogu biti bilo kog tipa tj. one mogu predstavljati sadržaj bilo koje klase objekata.

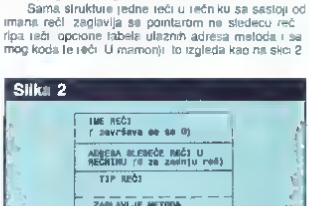
Ovo su ukratko bile osnovne operacije sa stekom. Pošto je *Manevro* potpuno otvoren sistam za nadogradnju nije teško definisati još neke reč za manipulaciju stekom ukoliko se pojavi potreba za njima

REČNIK

U *Manevro* sistemu sve deklaracija izvršnih reči (procedura metodi – koji ce bili objašnjani kasnije – varijabla i konstanta) se nalaze u posebnoj jednostruko ulancanoj listi nazvanoj rečnik Tri glavne operacije sa rečnikom su dodavanje novih reči (bice prazen hrano u sledecem poglavlju) brisanje reči iz rečnika kao i pregled do log trenutka definisanih reči

Sama struktura jedne reči u rečn ku sa sastoji od imena reči zaglavlja se pointarom ne sledecu reč tipa reči opcione tabele ulaznih adresa metoda i sa mog koda te reči *U memonji* to izgleda kao na slici 2

Slika 2



Struktura reči u rečniku

Promenljiva *hara* je usko povezana sa pojmom rečnika. Na toj adresi sa nalazi WORD koji pokazuje na adresu prve neiskorištene memorijske lokacije u sistemu. Ij na prvu slobodnu lokaciju posla rečnika. Od le adrese *Manevro* počinje da kompajlira nove reči

NOVE REČI

Definicija nove reči počinje deklarativnom rečju (dvotačka). Posle nje se navodi prvo ma nova reči zatim lalo procedure (sastavljeno od poziva ranije definisanih reči) a kraj se označava rečju (tačka zarez). Reči koje definiše korisnik su potpuno ravnopravne sa rečima koje *Manevro* zna u osnovnoj verziji

Prilikom redefinisanja neka reči prvo slovo stare reči se zamenjuje znakom ~ (tilde) tako da poslednja definisana verzija ostaje validna.

Kao što se može primetiti *Manavro* radi u dva modaliteta kao interpreter i kao kompajler. U interpreter modu (interaktivnom modu) sve što korisnik otkuca se odmah izvršava. Kada interpreter naiđe na deklarativnu reč tipa **const**, **var**, **method**, **class** ili **interrupt**, ulazi u mod za kompajliranje. Kompajliranje traje sve do znaka **;** ili do znaka **!ret** (za *interrupt procedure*) ili se završava odmah nakon uzimanja imena nove reči (slučaj sa deklaracijama **const** ili **varj**. Normalno kada je u kompajler modu *Manevro* ne dozvoljava još jedan ulazak u isti mod drugim rečima nije moguće direktno iz procedura definisati konstantu i varijablu. Napomenuto je da to nije direktno moguće ali su predviđene ključne reči **interpret**, **compile** i **make_proc** koje obezbeđuju nesmetano prebacivanje sistema iz jednog moda u drugi.

INTERAPT PROCEDURE

Razlika u ovom tipu procedura je to što se pozivu sve registri procesora čuvaju na povratnom steku. Početak kompilacije ovakva reči označava se pomoću reči **interrupt** (umesto **:**) a završava se rečju **iret**, umesto **;**. Ove procedure služe za pravljenje izv. TSR (*Terminate and Stay Resident*) programa za IBM PC. Naravno osim ovih uvedena su još neka reči za tu svrhu. **Init_m** inicijalizuje parametar stek a **keep** vrši izlaz iz *Manevro* sistema s tim da ceo paket ostaje u memoriji. Takođe obezbeđene su i osnovne operacije sa flagovima (*Carry* i *Zero*) kao što su **stc** i **clc** (*Set & Reset Carry Flag*). **ll c** (*If Carry*) koja radi isto kao i naredba **ll**, samo što na očekuja logičku vrednost na vrhu steka i **ll_z** (*If Zero*).

KONTROLNE STRUKTURE

Ovaj sistem je podržan standardnim setom kontrolnih struktura. Kontrolna struktura se pravodi isključivo u kompajler modu tako da se ne može napisati da radi u interaktivnom režimu.

Osnovna stvar za bilo koju kontrolnu strukturu je su konstanta **!true** i **!false**. U *Manevro* sistemu **!true** ima *integer* vrednost 1 a **!false** vrednost 0. Podržane su isključene reči (**!true** i **!false**) ze dovođenja ovih konstanti na vrh steka.

Osnovna struktura je svakako **ll_then_else** koja se shodno RPN notaciji ovde piše u malo izmenjenom radosledu. Reč **ll** sa vrha steka skida *integer* upoređuje ga sa 0. I ako je on različit nastavlja se sledećom instrukcijom. Ukoliko on jeste jednak 0 nastavlja se sa izvršavanjem prve reči navedana posle reči **else**. I ako na nema (znači konstrukcija tipa **ll_then**) posle reči **then**. Znači ovo nije konstrukcije **ll_then_else** nego konstrukcija tipa **ll_else_then**. Reč **then** uvek označava kraj ove konstrukcije (slično **endl** u nekim varijantama bejzika).

Sintaksa brojačke petlje je veoma slična sintaksi **for...nxt** (ili **FOR DO**) petlja. Ova petlja je malo neobične utoliko što svoje informacije (gornja granica, donja granica i indeks) čuva na povratnom steku. Reč **i** (slovo i) sa povratnog steka dobavlja trenutnu vrednost indeksa. Dodala su još neka reči podrška za ovu kontrolnu strukturu prevremeni izlazak iz petlje, opadajući brojač i brojač se promeni vim koracima.

U *Manavru* postoji još jedan tip petlje a to je **repeat...until** palja. Reč **repeat** na vrh povratnog steka stavlja samo adresu ulazne tačke petlje a kada se stam naiđe na **until** skida integer sa steka i upoređuje ga sa 0. Ukoliko jeste 0 viši se ponovan ulazak u petlju a ukoliko nije skida se zapslale ulazna adresa sa povratog steka i izvršavanje se nastavlja normalno. Vidi se da je ustov izlaska iz ova petlja često loe čke vrednost (**true** ili **false**) koja se nelazi na vrhu steka pia nailaska na kraj petlje.

METODI I KLASE

Razmotrimo sledeći primer treba implemantirati formulu

(A+B)*(C+D)

lako da ona ravnopravno radi i sa integeri brojevima i sa real brojevima a i sa malncama. U poznatim proceduralnim jezicima za svaki tip mora se pisati nova procedura. *Manavro* sistem prati sledeću logiku za različite tipove podataka imana operacije ostaju sta samo se menja njihovo dejstvo. Ukoliko imamo instalran tip *real* i *matrix* možamo pisati

```
$ ! formula a !a b !a + c !a d !a + *,
ok
$
```

i procedura formula će radili za svak od navedenih tipova.

Ovo je moguce ostvariti posebnim vrstama procedura nazvanim *metodi*. Metodi za razliku od običnih reči na svom početku imaju tabelu ulaznih adresa. Šta se desava prilikom poziva metoda sistem prvo pročita tip objekta sa vrha steka a zatim nađe ulaznu adresu procedure (sadržanu u tabeli ulaznih adresa) koje obrađuje taj tip objekta. Tima se omogućava da isle procedura iede nad svim tipovima ukoliko su od govarajuće operacije definisana.

Pošto ima ukupno 255 tipova tabela ulaznih adresa sadrži mesto za adresu za svaki od tipova. To znači da je tabela dugačka 512 bajtova a nalaženja ulazna adresa za određeni tip je sasvim jednostavno:

Adresa Ulazne Adrese = Adresa Početka Tabele + 2 * Tip

Ukoliko metod za određenu klasu nije implemanli ran u tabeli se nelazi vrednost 0 (kao ilklivna ulazna adresa) i prijavljuje se greska *Method Not Defined*. Sva ove operacije se izvršavaju lokom izvršavanja maloda a na lokom kompajliranja lako da sistam u loku tada određuje u zavisnosti od trenutnog tipa podatka koja će se akcija izvršili.

NOVE KLASE

Manevro obezbeđuje deklarativnu reč **class** koja služi za upoznavanje sistema sa novom klasom. To je u stvari deklaracija konstanta čiji će sadržaj biti identifikator (tip) te klase. Tip nove klase se definiše kao poslednji iskonštani tip uvecan za 1. Inace u originalnoj verziji u *Manevro* je instalirana samo jedna klasa (klase **integer**) i njan tip je 0.

Definicije novog metoda je vrlo slične definiciji procedure. Umesto reči **:** koristi se reč **method** koja na vrhu steka očekuje identifikator klase (tip) za koju se metod definiše. Ukoliko ime metoda na postoji u rečniku formira se nova reč zajedno sa tabelom ulaznih adresa a kod novog metoda sa smešta odmah posle tabele.

Ukoliko metod već postoji u rečniku (nađeno je da postoji reč se istim imanom) ne formira se nova reč u rečniku nego se popunjava tabela ulaznih adresa ranije definisanog metoda.

Sva do sada rečeno se može ilustrovati sledećim primjerom

```
$ class Moja_Nova Klasa
ok
$ Moja_Nova Klasa method :
$       <kod_meloda>
$ ;
ok
```

što će implamentirati operaciju + u klasi **Moja_Nova Klasa**.

KREIRANJE OBJEKTA

Kao što smo videli objekat na steku se sastoji od podataka dužina i identifikacionog broja klase. Najjednostavniji malod da se na stek ostavi neki objekat jeste da se on kra re baš po bajt-no. To je dosta zamoran posao. Isime kre anje objekta se na može izbeci (postoji nakoliko specijalno dizajniranih reči za kreiranje objekala na stekuj ali se može automatizovati leko da kompajler kada dođa do odredanog lor mala zapisa može to pretvoriti u konstantnu objekat određene klase. To se postiže definisanjem posebnog metoda nazvanog **get_c** (*get constant*).

Pomoću metoda **get_c** kompajler interpretira slu ži za prihval konstantnih objekata neke klase. Evo kako će kompilacija (ili interpretacija) u *Manevro* sistemu prvo se pretraži rečnik i ustanovi se da li ta reč već postoji. Ukoliko postoji ona se izvršava ili se kompajlira poziv te reči na prigodnom mestu. Ukoliko ta reč na postoji *Manevro* ulaz u petlju poziva **get_c** metoda. Poziva se prvo metod čiji rašnji broj klase je 0 (to je integar) zatim 1 itd. Metod **get_c** ima za zadatak da prapozna nepoznati tekst ukoliko se radi o nekoj konstanti. Recimo isti malod za klasu **integera** provaiava da li je neki zapis broj i ako jasta vraca sistemu vrednost tog broja. Ukoliko nije obaveštava sistem o lome pa on nastavlja sa pirolvkom metoda.

Omogućeno je da korsnik sam definiše metod **get_c** koji će prihvatati konstantne objekte. Ulazni parametal ovog malode je adresa teksta koj. *Manevro* sistem nije prepoznao kao ranije definisanu reč i koja se nalazi na vrhu parametalar steka. Ukoliko metod prapozna konstantni objekat reč **!last** obaveštava si stam gde je kraj nepoznatog teksta. e na vrhu se nalazi konvertovani objekat. Ukoliko teksl n-je prapoznel adresa početka se ostavlja na vrhu steka a si stem sa obaveštave o lome instrukcijom **clc**. Na listingu 1 je dat primer kako *Manevro* mozemo obogatiti lako da na ulazu shvata i heksadekadne brojeve kao brojeve koji počinju znakom $.

(Programen koj su bliski sa *Forth* jezikom će ve činom u potpunost razumeti ovaj primer. oni koj nisu uskoro će sa SEZAM a moć da prauzmu kompletan program i uputstvo za rad se *Manavro-m*.)

Pozlv metoda **get_c** iz vašeg programa ustovljave da objekat koji sa nalazi na adresi koju dostavite metodu bude pretvoran u konstantn objakat neke klase. Ukoliko laksi koji zelite da pretvorile na predstavlja objakat ni jedna klase prijavljuje se greška **Method Not Defined**.

Objekti svih klase mogu počinjati bilo kojim slovom ili znakom osim sledećih: **\ ' [ ]**. Takođe mora se voditi računa o lome da ovaj metod u slučaju prapoznavanje objekta obavezno oslievi na vrhu steka njegovu vrednost sa lačnom dužinom i tačnim identifikatorom klase.

NASLEDIVANJE

Iako je *Manavro* po dalinoji klasno baziran sistem to ne znači da na postoji mogućnost nasleđiva nja među klasama. Pretpostavimo da imamo opštu

```
Listing 1

0 var h_addr
0 var h_base
0 var h_flag
0 var h_len
$ var Hex_Limiter
0 var h_res

: Hdon      \ { ii    10 ii 1 dtbpedovanja broja 10
  dup 0 < if drop 1 else 10 swap ?   !Hex ! then

;

class Hex

Hex method get_c      \ jedini metod koji nam treba!!!

  dup get_byte hex_limiter @ =  \ da li je prvo slovo 0?
  if
    1 + h_addr !  \ ako jeste  pocetak konverzija
    4000 h_base !
    0 h_len !
    0 h_res !
    false h_flag !
    repeat
      h_addr @ dup 1 + h_addr !
      get_byte
      dup dup 39 - swap 10 - or
      if
        drop
        true h_flag !
      else
              dup 30 > if 213 and then
              dup  dup
              0 < if error Hex_What?? then
              dup 10 <
              if
                h_base @ * h_res @ + h_res !
              else
                7  dup 10 -
                if
                  h_base @ * 0 max 0 + h_res !
                else
                  error Hex_What??
                then then
              h_len @ 1 + h_len !
              h_base @ 10 div dup 0 =
              if drop ! h_addr ! else h_base ! then
    then
    h_flag @
  until
  h_len @ dup 0 >
  if
    h_res @
    swap 4 swap   !len
    div
    h_addr @ !last
  else
    error Hex_What??
  then
  else
    clc \ ukoliko prvo slovo nije 0   obnovsti komandar
  then
;
```

Prlnaer upotrebe programskog jezika *Manavro*

klasu A i njenu podklasu B. Teorija kaže da bismo u klasno bazliranom jeziku sve metode za klasu B morali ponovo da definišamo. Međutim postoji jedan mali trik kojim se programar može poslužiti ukoliko su te prezentacije objekata klase A i B na staku slične, da linicija metoda za klasu B može biti kao na listingu 2

Listing 2

```
{listing 2

B method +       \ recimo da implementiramo metod +
<pretvori B u A>  \ objekat klase B pretvori u objekat klase A
                  \ ovo bode aktivira metod za klasu A
<Pretvori A u B>  \ rezultat je formata klasa A
                  |
```

Primer definicija metode za klasu

Procedure **<pretvori X u Y>** i ne moraju postojati ako su reprezentacija objekata među klasama identične (recimo klasa matrice i klasa kvadratne matrice)

KOMPILACIJA I INTERPRETACIJA

Na ovom sistemu izgrađena su tri veoma interesantne reči koje omogućuju korisniku da prebacuje *Manevro* iz interpreterskog u kompilacioni mod i obrnuto iz svog programa. To, praktično, znači da je se svim tegalno napisati reč iz koje se definiše nova reč. Primer koji to najbolje ilustruje jeste definicija višedimenzionalnih nizova: gde poziv reči koje predstavlja identifikator višedimenzionog niza prvo odradi račune nje adrese određenog elementa

Prva reč **_Intalpret** nalaže *Manevro* sistemu da pređe u interpreterski režim. Na vrhu steka se očekuje adresa teksta koji će se izvršiti odmah, a koji je terminiran znakom ; (tačka zarez s). Iz ovog moda se mogu i kompajlirati reči tj. moguća je još jednom ući u kompajler režim

Sledeća reč je **make_proc**. Služi za formiranje zaglavlja reči negde u memoriji (na more u recniku) sve do prve instrukcije. Na vrhu parametar steka se očekuje adresa teksta po kojoj će reč biti nazvana i adresa od koje počinje reč. Uglavnom se koristi u paru sa rečju **_compile**. Ova reč izvodi kompilaciju zadog teksta od proizvoljnog mesta u memoriji. Pre poziva na staku se nalazi adresa teksta koji se završava znakom ; (tačka zarez) i adresa od koje počinje kompilacija. Na kraju reč **_compile** vraća prvu slobodnu adresu. Da bi se izvršila kompilacija u rečnik kao početno mesto za kompilaciju treba uzeti sadržaj promenjive **here**, a polom **here** kongovati na novu vrednost. Važno je reći da **make_proc** novu reč odmah ulančava sa zadnjom definisanom, tako da ma me bojazni de će rečnik izgubiti svoju funkciju

GREŠKE U PROGRAMU

Vrlo često se desi da reč koju ste napisali ne radi iz potpuno misterioznih i nerazumljivih razloga. U većini slučajeva krivac je sam programer i vrlo je zgodno obezbediti alat za traženje grešaka. *Manevro* ima tu mogućnost da u program koji se testira uključi informacije o izvršavanju. To se postiže rečju **/debug**. Posle navođenja te reči sva sledeća reč (koje se kompajliraju) će na početku imati kompajlirianu instrukciju **INT 63h**. Naprevljen je posebni mali debager program koji vam omogućuje *step* i *trace* kretanje kroz program, kao i pracenja sadržaja parametar-steka i proizvoljnih varijabli. Inače osnovne reči *Manevro*-a imaju poziv tog interapta na početku, ali u slučaju de je **/debug** postavljen na OFF instrukcija **INT 63H** se preskače pri kompilaciji

Prijavljivanja grešake u *Manevro* sistemu je izvebeno lako da se posle pojave greške program zaustavlja navodi sa uzročnik između navodnika. Ispisuje sa tekst greške zatvargju se eventualno otvoreni fajlovi (ovo važi za sistemsk otvorena fajlova ne za fajlove koji predstavljaju daloteke se podacima koje je otvorio sam programer) bnše se parametar i povratni stek i prelazi se u interaktivni režim, bez obzira u kom rezimu se sistem do tada nalazio. Naravno greška je moguće prijavih i iz vaše reči...upotrebom reči **error <Tekst_Greške>**. Ograničenja je to da se na može navesti šta je objekat greške (reč u navodnicima) i da tekst greška ne može sadržati blanko znaka

Podržana je i opcija da korisnik sam utvrdi akciju pn nastanku greške. To se obezbeđuje pomoću reči **on_err go**. koja očekuje adresu izvršne rutine koja će preuzeti posao *error handler*-a. Prvi deo ostaje isti, tj. ispisuje se tekst i broj se na parametar stek, a potom se kontrola predaje korisnikovoj rutini. Staro stanje stvari, tj. vraćanja na *Manavro* sistemsku rutinu za obradu greške se ostvaruje poziva poziva reči **restore_err**

Slika 4

```
$   vlist
$ asm{
$ code   segment
$        mov ax,#vlist
$        and ax,3
$        call ax
$ none   ends
$ end
$ },

Turbo Assembler  Version 1.01  Copyright (c) 1988, 1992 Borland International

Assembling file:   TEMP.ASM
Error messages:    None
Warning messages:  None
Remaining memory:  342k

Turbo Link  Version 2.0  Copyright (c) 1990, 1992 Borland International
Warning: no stack
ok
```

Povezivanje se asemblerom

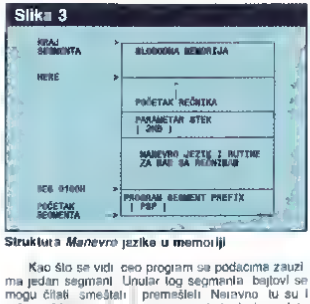
MAŠINSKA OKOLINA

Manevro u osnovnoj verziji nema nikakve specijalne instrukcija za rad sa I/O. Jedina dva reči koje se oslanjaju na DOS vašeg računara su **_** (tačka koja ispisuje objekat sa vrha steka (za početak *integer*) i uništava ga) i **load**. Reč **load <FileName>** radi redirekciju ulaza (tastatura) ka fajlu **FileName** i tim putem se sistem pun programima. Redirekcija ireje sve do reči **;** (tačka zarez s). Isto tako i u fajlu se može navesti **load**, sa neograničenom dubinom poziva. Sama reč **load** se izvršava isključivo u interpreterskom režimu rada (nije ja moguće kompajlirati)

Osim unapred definisanih poslopi dosta reči koje služa za povezivanje *Manevro*-a se MS DOS okolinom putem DOS interapt funkcija. Pre svega tu su reči **reg** (gde reg može biti ax bx cx dx si di es ds) koja postavlja sadržaj odgovarajućeg registra i reč **reg_** koja na vrh steka dovodi sadržaj registra (ovi sadržaji se koriste samo pn instrukcij **int**). Reč **int** izaziva softverski interrupt obično 33 koji je zadužen za DOS podršku. Isto tako tu su već pomenute reči za rad sa fllegovima koja omogucavaju delek tovanje grešaka koje prijavljuje sam DOS

Druga grupa reči odnosi se na manipulacije sa memorijom. Sama struktura *Manevro* jezika u memoriji je kao na slici 3

Slika 3



Struktura *Manevro* jezika u memoriji

Kao što se vidi ceo program se podacima zauzima jedan segment. Unutar log segmenta bajtovi se mogu čitati smeštati premeštati. Naravno tu su i reči za čitanje i pisanje po memoriji kada je zadata apsolutna adresa koja može biti bilo gde u memoriji vašeg računara. U sušini programer u osnovnom segmentu mora držati samo stek i rečnik. Prostor za podatke se može dinamički alocirati i nije ograničen na 64KB

Posebnu grupu procedura čine reči **lap** i **!a:sp** koja menjaju sadržaj pokazivača vrha steka. Drugim rečima moguće je ostvariti dva ili više parametar stekova ili potpuno relocirati postojeći

ASEMBLER

Kao što je napomenuto u ovaj programski paket je uvedena mogućnost da programer iz nekih razloga uključi asamblarski fajl u svoju reč. To se postiže naredbom **asm{** koja se ne može izvršavati u interaktivnom režimu (što je i log čno). Prilikom kompilacija posla ta reči *Manavro* sistem kreira fajl na lekućem disku pod imenom **temp.asm** i na početak log fajla piše tekst

ORG <adresa>

gde je **<adresa>** adresa od koje počinja kompilacija (trenutn sadržaj pokazivača adrese ne koji se vrši kompilacija reči). Posle toga se prepisuje sadržaj koji ja programar napisao ave od znaka }. Izuzetak od tog pravila čine reči koje počinju znakom * iaz mak(hash). Tada se čita reč koja dolazi posle tog znaka prelrazuje se rečnik i ukoliko se fakva reč nalazi u rečniku njen naziv se zamanjuje adresom zadnjeg slova njenog imena (ista adresa koju daje i reč **find_w**). Ukoliko nije nađena odgovarajuća reč u rečniku cela reč posle znaka * (zajedno sa njim) se prepisuje u fajl kao da se ništa nije dogodo lo. Time je obezbeđena minimalne ali efikasna seradnja asemblerskih fajlova se *Manavro* rečnikom

Posle kreiranja fajla **temp.asm** poziva se eksterni asembler i linkar. Komandna linija asemblera se sastoji od imena **temp asm** (proverite da i vaš asembler podržava parametre u komandnoj liniji), a istomena linije za poziv linkera (isto eksternog) se sastoji od imena **temp obj**. Posle pozivanja asemblera i linkera poziva se DOS program **exe2bin** koji **exe** fajl pretvara u formu pogodnu za uključivanje. Fajl **temp bin** (nastao posle **exe2bin**) se uključuje u beli nicju reči bnše se svi fajlovi osim fajla **temp asm** (koji ostaje radi provera eventualne greške) i nastavlja se normalno sa kompilacijom reči. Sve ovo ilustruje primar na slici 4

U slučaju greške na bilo kom mestu (greška u asemblerskom fajlu namogućnosi konverzije u BIN fajl i sl.) prijavljuje se greška **Error in Assembler File** bez oznaka gde je la greška nastala. Greška se detektuje pregledom fajla **temp asm**

Od procesorskih registara *Manevro* koristi samo BP koji pokazuje na trenutni vrh parametar steka. Ovaj registar se može menjati u okviru asemblerske rutine ali pn izlasku iz reči mora imati odgovarajuću vrednost. Inače na adresi **byte ptr [bp]** nalazi se tip klase a na adresi **byte ptr [bp-1]** dužina objekta. Pošto je sam *Manavro* **com** fajl registri CS, DS i ES imaju isiu vrednost (koju daje reč **myseg**)

Ceo sistem se asemblerom je zamišljen pre svega kao ispomoć pn pravljanju nakih reči koje je jednostavnije uradili u asembleru nego u bilo kom drugom jeziku. Isto lako na ovaj način ja omogućeno bekonsrnik proizvoljno implementira reči sa efektima koji se na mogu izvesti iz samog *Manevro* sistema kao što je rad sa mamonjskim blokovima, rad sa proširenom (produženom) memorijom, specifičnim I/O uređajima i slično

Rad na ovom projektu svakako nije okončan. Što konsnicima obezbeđuje još dosta novih modula klasa i reči ze razna primena. Posebna pažnja se usmerava ke nalaženju i ispravljanju bagova programa koji ne žalost sigurno ima (bilo bi ludo nadati se da ih nema). ali koji do sada nisu primecani. Zato se autor unapred zahvaljuje za svaku prijavljenu grešku u radu kompajlera i za svaki savet koji bi mogao pomoći da ovaj sistem pruži upravo ono što očekujete od jednog programskog jezika

OBJEKTI U PRAKSI

Čini se da su sve dobre ideje u istoriji bile otkrivane više puta. Žalosno je da čak i danas, uz sve informacione sisteme kojima raspolažemo, devedeset posto svega novoga ispadne „otkrivanje rupe u saksiji". Setite se toga siedeći put kada budete smišljaii neki novi algoritam.

Dejan Jelović

S obzirom na trenutno stanje jugoslovenske privrede, mislim da neću pogrešiti ako kažem da većina programera zarađuje za život na bazarne podelake. Kako tej posao ne razvije možđane vijuge baš previše, mnogi ljudi pronalaze još neku oblast kojom se bave kao nekom vrstom hobija. Kriptografija, algoritmi za sortiranje i pronalaženje slučajnih brojeva predstavljaju najzastupljenije razbibrige, mada se i moja oblast, ubrzavanje grafičkih algoritama, nalazi pri vrhu top-liste.

RUPA U SAKSIJI

Među grafičkim primitivama Bezierova kriva, zbog svojih karakterističkih, drži istaknutu poziciju. Za one koji to ne znaju, Bezierovu krivu je ranih sedamdesetih godina otkrio francuski matematičar Pjer Bezier, koji je u to doba radio za „Reno", pa mu je trebala kriva kojom bi opisivao karoserije automobila u tri dimenzije. „Računari" su već dosta pisali na ovu temu, tako da vas time neću davati. Ukratko, ukoliko imamo četiri tačke (u prostoru ili u ravni – broj dimenzija i nije bitan), Bezierovu krivu ćemo dobiti po formuli

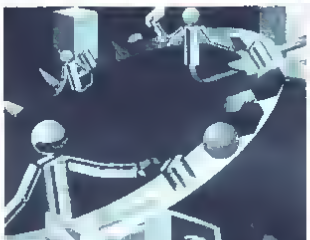
$$x(t) = t^3x_1 + 3t^2(1-t)x_2 + 3t(1-t)^2x_3 + (1-t)^3x_4$$

gde se t kreće od 0 do 1. Zamenjivanjem x sa y u ovoj formuli dobićemo, naravno, i y koordinate za našu krivu. Te je onda možemo prikazati na ekranu. (Formula koju sam dao jeste samo specifičan slučaj Bezierove krive u kubičnoj formi. Postoje i Bezierove krive koje se zasnivaju ne više od četiri tačke, ali se one mnogo manje koriste. Ova je prave sreće.)

Ono što Bezierovu krivu čini zanimljivom je to da kreće iz tačke (x_1, y_1), završava se u (x_4, y_4), a (x_2, y_2) i (x_3, y_3) su kontrolne tačke – kriva je u početnoj tački tangentna u odnosu na (x_2, y_2), a u krajnjoj u odnosu na (x_3, y_3). To je zgodno, jer sada lako možemo da spojimo dve Bezierove krive, namestimo početnu tačku druge krive na krajnju tačku prve i spojili smo krive. Sada još namestimo okolne kontrolne tačke, lako da zajedno sa tačkom u kojoj su krive spojene budu kolinearne i naš spoj će biti „gladak". I, na njemu neće biti nikakvog špica. Ovo svojstvo je toliko korisno da su gotovo sve firme koje se bave izradom vektorskih fontova počele da ih zasnivaju isključivo na pravim linijama i Bezierovim krivim.

Elem da se vratim na temu. Bezierove krive se zbog svog ovog stepenovanja i množenja crtaju veoma sporo. Postoje, naravno, postupci kojima se sve to svodi na jednostavnije operacije i uklanje se potreba za konačnjem realnih brojeva (tj. se kreće od nula do jedan), ali je sve to i dalje **sporo**. Kako se Bezierove krive konste u manje više svim danas raspoloživim fontovima, onoga ko bi pronašao brži algoritam za njihovo crtanje očekivala bi gomila zelenih šuškavaca. Mlad i zelen ja sam mislio da imam rešenje. Budala.

Moja „revolucionarna" ideja bila je da prvo na osnovu četiri tačke zamislim tri linije z_1 do z_2, z_2 do z_3 i z_3 do z_4. Onda bih, na osnovu te tri linije, takonstruisao još dve – od sredine z_1-z_2 do sredine z_2-z_3, i od sredine z_2-z_3 do sredine z_3-z_4. Kada odbacimo višak (linije vezane za z_2 i z_3 sada postaju nepotrebne), onda opet ponovimo postupak za svake dve susedne linije, spajamo njihove sredine i onda odbacujemo onu tačku koja ih je spajala, kao i njene dve „paralelske" polovine linija. Ponavljajući ovaj postupak pet šest puta, dobili bismo sasvim dovoljno tačeka da nacrtamo krivu. Ne moje iznenađenje, ovaj algoritam je zaista crtao Bezierove krive i to praktično dvapul brže nego original. Sve one matematičke operacije su zamenjene samo nalaženjem tačke tačno između druge dve. Što se svodi na sabiranje i šiftovenje. Drastično ubrzanje!



Opijen oduševljenjem, brzo sam poslao pismo ljudima koji se razumeju u problematiku. Oko mesec dana kasnije, dobio sam i prvi odgovor. Ako isečemo časkanje i formalnosti, ono što me želeo sl preostaje je ovo

U svekom slučaju, čestitamo ti, Dejane. To što si otkrio je pojednostavljene verzije onoga što se naziva „de Casteljau algoritam". Konsti se za polovljenje Bezierovih krivih. Ponavljanjem procesa polovljenja, dolazi se do krive koja je tako mala da se samo može povući linije između četiri tačke, što mi upravo i radimo.

Tvoja verzija algoritma ima jedan nedostatak: s to je da se krajnje tačke krive zanemaruju tj. imaju oko sebe mnogo manje određenih tačaka nego sredina krive. Dok je ovo prihvatljivo kod samostalne krive, kada treba da se poveže nekoliko krivih, dolazi do lošeg uklapanja. Žao nam je.

Ode moj milion dolara. Šmrc.

KO UBIJA OBJEKTE?

Gornja priča (s tužnim krajem?) nosi lepu poruku, pre nego što počnete da se ozbiljno bavite bilo čime, dobro bi bilo da pročitate na koliko knjiga ne tu temu. Niko na sumnja da sve možete da shvatite i sami, ali – bolje je učiti na njihovim greškama nego na vašim. Koju god oblast da izaberete, neko je na nju već potrošio deseak godina. Ukoliko taj neko nije baš potpuni idiot, imaće bar nekoliko dodnh saveta koji će vam uštedeti dosta vremena i time isplatiti vreme utrošeno na čitanje njegove knjige. Najčešće to zapravo i nisu pogrbljeni meton proleson matematike koji pišu programe na papiru već programeri kao i vi koji su to pisali na osnovu praktičnih iskustava.

U tom pogledu nema razlike ni u oblasti objektnog programiranja. Ljudi se time bave već dvadesetak godina i prvo treba da vidite šta su drugi po tom pitanju uradili. Ukoliko koristite Turbo Pascal ili C++, nije dovoljno da pročitate knjige koje se bave isključivo ovim jezicima, već treba da potražite i one koje pokušavaju da odjasne kako se pišu objektni programi. Evo o čemu govorim.

Automatsko čišćenje *memorije* predstavlja jedan od sedam „zlatnih" ciljeva koje svaki OOP jezik treba da ispunjava. Uprkos tome, gotovo nijedan hibridni jezik ga ne podržava, jer bi to usporavalo program. Da bi se ostvarilo čišćenje memorije, svaki program bi morao da ima interapt rutinu, koja bi se javljala svakih nekoliko sekundi i koja bi

- Prošle kroz ceo *heap* i uzela podatke o svim alociranim blokovima
- Prošla kroz slek i ispitala pointere ne njemu da li pokazuju na neki od ovih blokova. Ukoliko pokazuje, taj blok bi bio markiran. (Za prevo bi rutine ispitala *sve* podatke na steku, jer nema načina da utvrdi koji je pointer e koji, recimo, int.)
- Prošla kroz markirane blokove i za sve pointere u njima opet markirala blokove na koje pokazuju
- Obrisale sve nemarkirane blokove

Sve ove operacije, naravno, uzimaju procesorske cikluse i veoma se teško „kaleme" na klasičan jezik kakav je C ili Paskal, pa su zato tvorci hibridnih jezika odlučili da ih ne implementiraju. Nema veze, kažete vi, nije meni teško da za svaki objekat koji sam alocirao na *heap*-u uradim **delete** E, u tome je stvar! Nije ni njima bilo teško. Problem je, naime, na sasvim drugoj strani.

Recimo da imate klasu **List** koja predstavlja uvezanu listu u koju stavljamo stringove tipa **String**. Neke od tih stringova stavljamo da bismo ih odložili i više ne razmišljamo o njima, a ne neke i pored toga što su u listi i dalje imamo uperene pointere i koristimo ih u drugim delovima programa. Problem će nastati kada poželimo da uništimo listu, bilo zato što je taj objekat izašao iz funkcije u kojoj je deklarisan, bilo zato što mi to radimo komandom **delete**. Ukoliko lista nije prazne, šta treba uraditi da bismo je uništili? Da li da prelpostavimo da za svaki objekat u njoj programer ima referencu te da će se sam pobrinuti o tome da ih izbriše, ili da ih sve od reda obrišemo, uzkjučujući da neki drugi deo programa o tome nema pojma pa da preko pointera referencira objekat koga više nema? U sistemima sa automatskim čišćenjem memorije, mi o tome ne bismo brinuli. Listu bismo jednostavno uništili, a mehanizam za čišćenje bi sam „ubijao" objekte kada ne njih više ne postoje reference.

Naravno, iako je meni najlakše da objasnim

Listing 1

```
{
    Menu menu,
    Choice file ("File", filemenu),          // na steku

    menu + file
         + "new Choice ("Edit", editmenu)  // alocirane na hrapu

}                                          // Ovde nastaju problemi
```

Listing 2

```
int isOnHeap (void *object)

{
    #ifndef __MSDOS__
        #error This routine works only on MS DOS machines
    #endif
    #ifdef __TINY__
        #error Tiny model not allowed for this function
    #endif
    #ifdef __SMALL__ || __MEDIUM__
        return object < &object
    #else
        return (unsigned long)FP_SEG(object)<<4) + FP_OFF(object) >
            (unsigned long)_SS << 4) + _stklen,
    #endif
}
```

Listing 3

```
class vector3D {
    float v1  v2, v3;
    friend ostream& operator << (ostream &astr  const vector3D v)
        { return astr <<  (  << v.v1 <<  ,  << v.v2 <<      << v.v3 <<  ) ; }
};
```

ovaj problem baš na nekoj kontejnerskoj klasi (list, **array**, **bag** – objekti koji su zaduženi da sadrže druge klase); ovaj problem se može javiti svuda. Je sam u svom programu imao klasu **Menu** i njen objekat **menu**, na koji je trebalo da „dodajem" pojedinačne stavka u menijima. Međutim trebalo je da neke stavke dodetno kontrolišem, pa sam imao i dodatne reference na njih. To je izgledelo otprilike kao na listingu 1.

Kada bih na kraju funkcije uništavao objekat **menu** ukoliko bi **menu** automatski uništavao sve stavke (klase **Choice**) pokušao bi da sa **delete** uništi i stavku **file**, koje je ne steku. Sa druge strane, eko ih ne bih uništavao, onda bih, da bih mogao ručno da je uništim, morao i za stavku koja vodi u **editmenu** da držim pointer, iako mi on ne treba. Trulo.

Ako hoćete neko „normalno" rešenje, je ga nemam, a mislim da i ne postoji. Ono što sam uz pomoć Zorana Životića implementirao kao neku vrstu zakrpe, jeste funkcije koja određuje da li je neki objekat na *heap*-u ili na steku (listing 2).

Uz to sam sve kontejnerske destruktore izmenio, tako da uništavaju isključivo objekte na *heap*-u, i uvao konvenciju da sve objekte koji treba da imaju još neku referencu alociram na steku. Ovo radi savršeno, uz dva nedostatka: prvo, užasno je neprenosivo i drugo, ned o bog da neko ko ne zna za ovu konvenciju uzme da konsti ove klase. Mirisao bi ljubičice odozdo. Bio bi mrtav. Dead. Kaput.

ŠTA JE VIRTUAL INLINE?

Svaki objekat ima uz sebe i nešto što se zove „pokazivač na tablicu virtuelnih funkcija". U toj tablici se nalaze, redom, adrese svih virtuelnih funkcija koje figurišu za tu klasu. Svaki put kada treba da se pozove virtuelna funkcija za neki objekat, njena adresa u memoriji se potraži u ovoj tablici. (Tablica je odvojena od objekta, jer je, zapravo, potreba samo jedne tablica za svaku klasu. Bilo bi to zaista traćenje memorije kada bi hiljade objekata iste klase nosili svaki svoju tablicu.)

U takvim uslovima *virtual inline* funkcije zaista zvuče kao loša šala. *Inline* funkcije ne stoje nigde već se po potrebi ubacuju direktno u kod, a virtuelne funkcije moraju negde da budu smeštene da bi tablica virtuelnih funkcija mogla da pokazuje na njih. Ipak, to nije apsurd. Virtuelne *inline* funkcije se zaista tretiraju kao prave virtuelne funkcije, tj. biva im dodeljeno mesto u memoriji. Što se tiče „inline" delaona je u akciji samo ako implicitno odredite klasu iz koje vam funkcija treba. Znači, ako kažete recimo

funkcija (5),

biće pozvana virtuelna funkcija. No, eko kažete

klasa::funkcije (5),

ona će biti upisana **inline**. Pri tom, naravno, ne treba zaboraviti da u vreme izvršenja konstruktora nijedna funkcija ne biva izvršena kao virtuelna, jer tablica virtuelnih funkcija u tom trenutku još nije napravljena, ta je to mesto gde će ovakve funkcije definitivno biti pozvane kao **inline**.

Ako se malo zamislite nad ovim mehanizmom virtuelnih **inline** funkcija nameće se pitanje: a gde onda biveju u programu zapisane te funkcija, ako negde moraju da budu zapisane da bi bile virtuelne? Za razliku od „normalnih" funkcija, koje se definišu samo u jednom modulu, **inline** funkcije se najčešće definišu u heder-datotekama, koje su na raspolaganju svim programskim modulima. Da li onda kompajler ubacuje definiciju **virtuel inline** funkcija u svaki modul? Odgovor je, naravno, ne. Ljudi koji su se bavili teorijom objektnog programiranja su (i) ovo predvideli, te je još odavno odlučeno da definicije **virtuel inline** funkcija budu smeštene uz proizvoljnu (kompajler sam bira) funkciju koja nije **inline**. Ipak, ni kompajler nije svemoćan. Ukoliko imate klasu u kojoj su sve funkcije **inline**, onda će, ne znajući gde će ih smestiti, **virtuel inline** funkcije ubaciveti najverovatnije u svaki modul zajedno sa tablicama virtuelnih funkcija. Zato, ukoliko imate neku klasu sa isključivo **inline** funkcijama, bilo bi dobro da ubacite i neki „ne-inline" član, i time uštedite dva-tri kilobajta.

NIKO NE SME DA IH DIRA

Ukoliko niste primetili, u prošlom broju „Računara" se pojavio jedan napad na moj život i delo. Ne bih ves sada uvodio u detalje te velikansko komunternovske zavere (ko hoće može da pročita moj odgovor negde u ovom broju), ali bih vam preporučio da pročitate to pismo zbog nekoliko zanimljivih rečenica koje otvaraju mali prozor u izuzetno zanimljivu disciplinu dizajna objektnih programa. Mene je, opet, to pismo podsetilo da u „Računarima" još uvek nije pala dobra diskusija na temu **friend** klasa i funkcija. Hajde da to odradimo.

U prošlom broju smo se bavili enkapsulacijom podataka unutar klase: te neću time opet da davim. Ukratko: za sveku klasu treba jasno definisati ono što je privatno (to samo ona sme da dira), ono što je zaštićeno (to smeju dirati i nasledníci) i ono što je javno, što svi smeju da diraju (pričamo o podacima – devojke koje ste nekada znali nemaju se tim veze). **Friend** klase i funkcije predstavljaju način da se ovaj mehanizam zaobiđe tj. time što neku klasu ili funkciju proglasimo za **friend**-a, mi joj dopuštamo da dira ček i privatne delove neke druge klase. Ništa loše u tome.

Ono što jeste problem je to što mnogi ljudi slijepo **friend** mehanizam. Umesto da lepo isplaniraju kako mogu koristiti i proširivati klasu, pa da na osnovu toga odrede koji članovi treba da budu javni koji zaštićeni a koji privatni, oni to urade „po osećaju" i zeznu stvar. Kada im mesec dana kasnije zatreba neka „private" promenljive, da se ne bi mantretirali, oni samo pozovu zaglavlje za tu klasu i ubace **friend** deklaraciju. Bez veze. Malo planiranje daje odlične rezultate i (koliko to puta ponavljam) zapravo štedi vreme.

A to sto se matematičanme i drugim teoretičanma **friend** klase i funkcije baš ne svidaju mnogo me ne uzbuduje. Nije teško zamisliti situaciju u kojoj je **friend** klasa izuzetno korisna. Na primer, imate objekat čijim pojedinim članovima treba da ime pristupa konačan broj klasa (objektni sistemi nemaju vrhe – teoretski mogu da sadrže beskonačno mnogo klasa), a da ostale klase ne treba da imaju nikakvo pravo pristupa. Bilo bi opasno te članove proglasiti javnim, s druge strane, eko bi oni bili privatni, klase koje bi trebalo da imaju pristup ne bi im mogle pristupiti. Rešenje bi se naravno moglo naći u eksplicitnoj proveri lipa podataka, ali to bi bilo silovanje po svaku cenu – **friend** klase su najnormalnija stvar u takvim situacijama.

(Uzgred: ukoliko treba da obezbedite I/O vaših objekata, ne morate da za svaku novu klasu koju napravite proširujete klase **istream** i **ostream**. **Friend** funkcije će vas u tom slučaju izvuci mnogo bolje (listing 3).

Pri tom, zapamtite: **friend** funkcije su koncepcijski mnogo manje vredne (jer ne mogu biti virtuelne) od funkcija članica. Upotrebljavajte ih samo tamo gde morate.

TELEGRAMI PODRŠKE

Cela ova stvar sa kritikom mog teksta me je podsetila na staru pnču o tome da li treba objavljivati adrese autore u „Računarima" radi kontakte sa čitaocima. Od toga se manje-više odustalo iz nekih tehničkih razloga, ali sade mi se opet čini aktuelnim, pa ako neko hoće da kontaktira sa mnom, može to da obavi na adresu: Petrijanska 70, 11090 Beograd, ili me može naći pod BUEF78 JELOVI050900 na Decnetu i DJELOVIC na Sezamu. Sve kritike, pretnje i ucene i dalje šaljite na adresu redakcije, ja pnmam samo krupne apoene stranih valuta, ključeve od skupih kola, telegrame podrške. I pisma vezana za C++ i objektno programiranje.

ADA computers

BEOGRAD
Tadeuša Košćuškog 72
tel/fax: 011/186-267; tel:011/186-355

NOVI SAD
Sirška 42
tel/fax: 021/416-189

MODEL 286/20 MHz	1 190
MODEL 386SX/33 MHz	1 390
MODEL 386/40 MHz/64K CACHE	1 790
MODEL 486/33 MHz/256K CACHE	2 990
MODEL 486/50 MHz/256K CACHE	3,490

Osnovne konfiguracije sadrže
40 Mb HOO, flopi disk 1,2 Mb, tastaturu, desk top kućište, monitor mono herkules 14" crno beli, 2S/1P port
model 286 1 Mb RAM, 386SX 2 Mb RAM, 386 i 486 4 Mb RAM

DOPLATE

HARD DISK 80 Mb	130
HARD DISK 105 Mb	190
HARD DISK 120 Mb	240
HARD DISK 210 Mb	490
VGA MONO MONITOR 14" SA VGA KARTICOM	130
KOLOR MONITOR SVGA 1024 X 768 14" 512 kb	430
KOLOR MONITOR SVGA 1024 X 768 14" 1 Mb	530
KUĆIŠTE MINI TOWER	20

OSTALA OPREMA

EPSON LX-400/800	480
EPSON FX-1170	1490
EPSON LQ-1170	1790
LASERSKI ŠTAMPAČ HP III P	3400
LASERSKI ŠTAMPAČ HP IV	4300
KABL ZA ŠTAMPAČ	30
YU SET ZA LX MODELE	50
KUĆIŠTE DESK TOP	170
KUĆIŠTE MINI TOWER	190
KUĆIŠTE MIDI TOWER	250
KUĆIŠTE TOWER	310
MATIČNA PLOČA 286-20	180
MATIČNA PLOČA 386SX-33	310
MATIČNA PLOČA 386-40 CACHE 64K	500
MATIČNA PLOČA 486-33 CACHE 256K	1450
MATIČNA PLOČA 486-50 CACHE 256K	1950
I/O KARTICA 1PAR/2SER	50
MGP (Herkules) KARTICA	50
VCA 16 BITNA 512K KARTICA	120
VCA 16 BITNA 1Mb KARTICA	250
AT IDE KONTROLER	50
KOMBI KONTROLER	100
1Mb RAM MODUL	100
256K RAM MODUL	25
TASTATURA KLIK 101	100
FLOPI DISK 1,2 Mb 5,25"	150
FLOPI DISK 1,44 Mb 3,5 "	120
HARD DISK 40 Mb	390
HARD DISK 80 Mb	520
HARD DISK 105 Mb	580
HARD DISK 120 Mb	630
HARD DISK 210 Mb	880
MONO MONITOR (Herkules)	270
VGA MONO MONITOR	300
KOLOR MONITOR SVGA 1024x768 14"	600
MIŠ SA PODNOŽJEM	60
ETHERNET KARTICA 16 BIT	290
STRIMER COLORADO JUMBO INTERNI 120 Mb	590
STRIMER COLORADO JUMBO INTERNI 250 Mb	690
STRIMER ARCHIVE INTERNI 150 Mb	1800
387-33 KOPROCESOR	250
MODEM INTERNI 2400	250
FAX/MODEM INTERNI 2400/9600	300
UPS 500VA	980
TONER ZA HPIII	250
TRAKE ZA STRIMER	100
DISKETE 5,25" 1,2 Mb	25
YU SET ZA HERKULES	30
FILTER ZA MONITOR MREŽNI	40

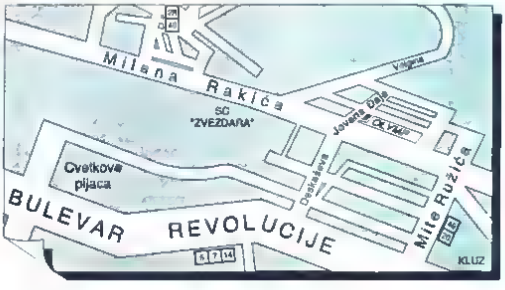
RADNO VREME OD 9 DO 17 ČASOVA, ISPORUKA ODMAH PO UPLATI
CENE SU U DEM, DINARSKA UPLATA PO DNEVNOM KURSU NA TRŽIŠTU U BEOGRADU
GARANCIJA 12 MESECI, SERVIS OBEZBEĐEN

Dosegnite OLYMP računarstva

OLYMP
electronic

tel. 011/400-477
fax 011/410-240

11000 BEOGRAD, Jovana Đaje 10



ADRESNI MODOVI

Pošto smo upoznali arhitekturu mikroprocesora 80486, vreme je da polako pređemo na njegov 32-bitni set instrukcija. Počećemo, kao i obično, od modova adresiranja...

Dejan Ristanović

U prošlim „Računarima" detaljno smo upoznali načine na koje mikroprocesor komunicira sa memorijom. Njihov zajednički sadržalac svakako je linearna 32-bitna adresa koje adresnom magistralom putuje od mikroprocesora do memorije. Što se hardvera tiče, ta linearna adresa je zbilja jedino što je potrebno da bi se pribavio (ili upisao) željeni podatak. Sa softverske strane, međutim, stvari su mnogo složenije – za programiranje i, nerocito prevodenje programa pisanih na višim jezicima potrebni su najrazličiji adresni modovi, od sasvim jednostavnog navođenja adrese do veoma složenih višestrukih indeksiranja. Iako nema sumnje da 80486 predstavlja veoma savremen mikroprocesor njegova arhitektura je klasično fon nojmanovska – instrukcije i podaci dele istu memoriju a programi se izvršavaju linijski, uz izvesno preklapanje izvršenja naredbe sa pripremom narednih. Što se obavlja potpuno transparentno za korisnika. Zato su adresni modovi prilično klasični i verujemo da će biti razumljivi svakome ko je ikada programirao na bilo kom „Intel"-ovom ili „Zilog"-ovom mikroprocesoru.

Na slici 1 prikazan je globalni način kodiranja instrukcija mikroprocesora 80486. Prvi bajt (ili prva dva bajta) su kod instrukcije. *r/m* i *s i b* (po bajt) opisuju modove adresiranja, *displacement)* je deo adrese i može se prostirati na 1, 2 ili 4 bajta, dok je *data* podatak iste dužine. Jedino je kod instrukcije obavezan – zavisno od tipa neke instrukcije neće uopšte imati adresu ili podatke. Instrukcije XCHG EAX, EBX na primer ima samo kod i *r/m* polje, dok su kod instrukcije ADD [EBP+2][ESI*4] 123 zastupljena sva polja sa slike 1. Kao dodatak, ispred nekih instrukcija pojavljuje se prefiks ili čak više prefiksa – neki veoma složeni oblici instrukcija mogu imati i četiri prefiksa!

U okviru ove serije lekcija nećemo se preterano baviti načinom na koji se instrukcije upisuju u memoriju pošto je taj način malo koga bilan – većina programera će pisati svoje programe na asembleru i onda ih primenom MASM-a ili nekog konkurentskog programa automatski prevesti u izvršni kod. Srećne okolnosti je što su razni komercijalno raspoloživi asemblerski programi za „Intel" ove mikroprocesore relativno dobro standardizovani tako da se sve bitne stvari unose na isti način što recimo nije bio slučaj kod mikroprocesora Z-80. Ako razlike i postoje one se svode na neke dodatne asemblerske direktive koje se i tu ko relativno retko koriste.

Implicitno adresiranje

Najjednostavniji adresni mod prisutan kod svih mikroprocesora je implicitno (podrazumevano) adresiranje, što znači da instrukcije nema adresu niti joj je ona potrebna. Postoji čitava klasa instrukcija koje obavljaju precizno definisan posao koji upošte ne zahteva komunikaciju sa memorijom – dobar primer bi bila instrukcija STI koja omogućava interapte ili CLD koja briše *direction* flag. Implicitno adresiranje se ne odnosi uvek na flegove nego i na sadržaje registara – instrukcija AAA na primer prilagođava sadržaj akumulatora posle ASCII sabiranja.

Registarsko adresiranje

Instrukcija AAA operiše sa sadržajem registre, ali je to tačno preciziran registar. Većina instrukcija, naravno, može da pristupa raznim registrima što znači da se u okviru same naredbe navodi na koje registre treba da deluje. To je registarsko adresiranje zastupljeno kod naredbi tipa MOV AX,BX ; INC ESI ; CALL EDI (poziv potprograma čija je adresa u registru EDI) ; ADD EDX, EBX itd. Na slici 2 prikazana je „mapa" registara opšte namene koje registarsko adresiranje „prepoznaje". Vidimo da su neki registri iz razloga kompatibilnosti podeljeni na polovine i četvrtine kojima se može slobodno pristupati. MOV EAX, EBX na primer prenosi 32 bita, MOV AX, BX 16 bita a MOV AL, BL samo osam bitе podataka iz registra EBX u EA.

Neposredno adresiranje

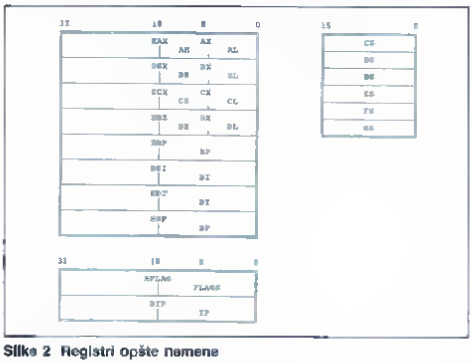
Kod neposrednog (*immediate*) adresiranja podatak se nalazi u okviru same instrukcije. Naredbe MOV EAX, 10 na primer prenosi konstantu 10 u registar EAX dok JC 3C0h preskače sledećih 960 bajtova ako je *carry* flag setovan. U instrukcijama naravno ne figurišu samo ovako „velike" adrese – neposrednim adresiranjem smatra se i instrukcija nalik na BT EAX, 3, koja kopira treći bit registra EAX u *carry*.

Kod primene neposrednog adresiranja konstante je pogodno izražavati u heksadekadnom obliku, ali se povremeno ukazuje potreba i za dekadnim i binarnim brojevima. Samom mikroprocesoru je to naravno potpuno svejedno jer sav posao obavi asembler – uobičajeno je da dekadne konstante prati sufiks D (npr. MOV AX, 123D), heksadekadne H a binarne B. Ukoliko se sufiks ne navede podrazumeva se D. Neki asemblери ugrađeni u više programske jezike mogu konsultovati drugu konvenciju – recimo da heksadekadnom broju prethodi prefiks $ ili &.

Specijalni oblik neposrednog adresiranja je adresiranje periferije. *I/O operands* u „Intel"-ovoj terminologiji. Mikroprocesor 80486 ima 65536 I/O portova, pri čemu je komunikacija se vrši 8-bitna, 16-bitna ili 32-bitna. 16-bitne i 32-bitne adrese moraju biti „uravnjene" na početak reči (deljive sa 2) odnosno duge reči (deljive sa 4). Na primer, OUT 06h,AL prenosi podatak iz AL na port 6h, dok IN EAX, DX prenosi u EAX 32-bita sa porta čiji je broj upisan u DX. Treba još znati da oznake porta ukoliko ih specificirale u okviru instrukcije ne smeju preći 255 – poriovima 256-65535 morale da pristupale indirektno kao u prethodnom primeru.

Dejstvo pojedinih IN i OUT instrukcija ne zavisi od samog mikroprocesora nego od konfiguracije sistema. Ij periferije koja je (ako je)

Slika 1. Kodiranje instrukcije mikroprocesora 80486



Slika 2. Registri opšte namene

```
brojač  db      10                      ; 32 bita za BROJAČ  inicija na vrednost  10
flag    dw      ?                       ; 16 bita za FLAG
ime     db      20 DUP (?)              ; 20 bajta za IME
        mov     brojač                  ; umanjuje BROJAČ za 1
        mov     AH  ime                 ; upisuje IME[0]  prvi bajt niza    u AH
        mov     AL, ime[1]              ; Upisuje IME[1]  drugi bajt niza) = AL
        or      flag  2000h             ; setuje bit u deliji FLAG
```

Slika 3. Direktno adresiranje

```
keyrd   STRUC                           ; definiše oblik sloga
        x       db      ?
        y       db      ?
        z       db      ?
keyrd   ENDS
ceKku   keyrd<>                         ; rezerviše memoriju za slog
        LEA     ESI,  tačka             ; adresa strukture u ESI
        MOV     EAX,  [ESI] x           ; x komponenta u EAX
        DEC     [ESI]y                  ; umanjuje y komponentu za 1
        MOV     [ESI] x,  [ESI]+z       ; kopira z u x
```

Slika 4. Adresiranje baza plus indeks - pristup strukturama podataka

```
        .list V[MAX_XLEN];
        .registar int i;
        suma = 0;
        for (i=0; i<MAX_XLEN; i++;  suma+=V[i]*V[i]

        pzrog:  XOR     ECX, ECX            ; brojač i (ECX) dobija vrednost 0
                MOV     suma, ECX           ; suma=0
        petlja: CMP     ECX, MAX_XLEN       ; da li je brojač veći od MAX_XLEN?
                JGE     kraj                ; ako jeste, kraj petlja
                MOV     EAX, ECX            ; Brojač u EAX
                SHL     EAX, 2              ; množi brojač sa 4
                MOV     EAX, V[EAX]         ; sadržaj niza u EAX
                CWDE,   EAX                 ; kvadrira sadržaj EAX
                ADD     suma, EAX           ; računa sumu
                INC     ECX                 ; uvećava brojač
                JMP     petlja              ; nastavak rada u petlji
        kraj:

        pzrog:  MOV     ECX, ECX            ; brojač i (ECX) dobija vrednost 0
                MOV     suma, ECX           ; suma=0
        petlja: CMP     ECX, MAX_XLEN       ; da li je brojač veći od MAX_XLEN?
                JGE     kraj                ; ako jeste, kraj petlja
                MOV     EAX, a[ECX*4]       ; sadržaj niza direktno u EAX
                IMUL    EAX                 ; dvadrira sadržaj EAX
                ADD     suma, EAX           ; računa sumu
                INC     ECX                 ; uvećava brojač
                JMP     petlja              ; nastavak rada u petlji
        kraj:
```

Slika 5. C program koji računa zbir kvedrata i dve moguca prevoda

```
Instrukcija                         Opis

ADD [ECX][EBP]  32                  EBP je bazni registar  koristi se SS
MOV AX  BX[EBP]                     EBP je bazni registar  koristi se SS
MOV EAX  [ECX][EBP*4]               ECX je bazni registar  koristi se DS
INC WORD PTR [ECX*8][EAX] X         EAX je bazni registar  koristi se DS
```

Slika 6. Korišćenje registra EBP kod edresiranja "Indeks plus ofset

```
Instrukcija                         Opis

MOV EAX  [EBP*8][EBX]               Matrica sa osam podataka sa EBP*4
INC WORD PTR [EDX*4+2]              Vektor iz bitih podataka sa EAX
MOV ECX  MOVE[EAX*8][EBX] Y         Niz struktura KoordX
```

Slika 7: Adresiranje "baza plus indeks plus ofset"

povezana se tim portom. Neopreznо slanje podataka na portove može da izazove nepredvidljive posledice koje obuhvataju „rušenje" sistema ili čak kvar na opremi. Zato većina modernih operativnih sistema (npr. OS/2) omogućava samo privilegovanim programima da direktno opsluže hardverom na ovaj način.

Direktno adresiranje

Sva prethodna adresiranja i nisu adresiranja u užem smislu je reči – kakvo je to adresiranje ako sa ne radi sa memorijom? Direktno (ili apsolutno - nadalje taj termin baš ne odgovara „intel"-ovoj segmentiranoj arhitekturi) adresiranje obezbeđuje komunikaciju između memorije i registara na najednostavniji mogući način instrukcijom MOV AL [12345Ah] na primer, prenosimo podatak iz ćelije čija je adrese 12345Ah u registar AL.

Ukoliko naredbu napišemo bez pominjanja segmenta ona se implicitno odnosi na segment sa podacima (*data segment*) na koji ukazuje register DS sa stike 2. Često treba pristupiti nekom drugom segmentu – u tom slučaju se piše ime njegovog registra i dvotačka na primer MOV AL, ES [12345Ah]. S obzirom da se ova instrukcija u mamorijl kodira u obliku prefiks+kod neki stan asemblen su zahtevali njeno pisanje u obliku

```
ES
MOV AL,[1235Ah]
```

pa je taj oblik prsutan i kod novijih programa tog tipa zbog potreba za kompatibilnošću. U praksi se međutim uvek piše **MOV AL, ES [12345Ah]**

Veličinom registra određena je zona memorije na koju instrukcija deluje – da smo koristili MOV EAX [12345Ah] bila bi prenesena 32 bita a ne samo 8 kao u našem primeru. Kod nekih isirukcija koje operišu direktno sa memorijom ova koncepcija ne daje rezultate – kako bi na primer mikroprocesor znao koliko bajtova treba da šiftuje kada bi naisao na instrukciju SHL [2000h]? Zato se u ovakvim slučajevima koristi BYTE PTR, WORD PTR ili DWORD PTR instrukcije SHL BYTE PTR [2000h] 3 na primer, šiftuje bajt memorije za tri bita ulevo odk bi INC DWORD PTR [2000h] delovalo ne samo na lokaciju 2000h nego i na sledeće posmatrane kao *longint* odnosno *double word*.

U praksi se apsolutna adresa veoma retko koriste pošto bi ucinile program mašinski zavisnim i teškim za održavanje. Uspešno ih zamenjuju labele. Primer sa slike 3 najpre reserviše 32 bita za polje BROJAČ zatim 16 bita za FLAG i najzad 20 puta po bajt za polje IME – prvom polju je dodeljena konkretna vrednost a ostala nisu inicijalizovana, što je opisano upitnikom (vrednost nebitna). Zatim se koriste instrukcije nalik na DEC BROJAČ ili MOV AJME[5] – daleko su preglednije od velikih nizova heksadekadnih cifara!

Bazno adresiranje

Kod baznog adresiranja adresa podatka se nalazi u registru – zelo se kod nekih drugih mikroprocesora ono naziva *indirektno* adresiranje (kao neka vrsta suprotnosti opisanom direktnom adresiranju). Instrukcija MOV AL [ECX] na primer, dovodi u AL 8 bita upisanih u memorijsku ćeliju *data segment*-a čija je adresa upisana u ECX. Ukoliko je na primer u ECX prethodno upisan broj 1234h MOV AL [ECX] bice akvivalentno sa MOV AL [1234h]. Ne ovaj način sa takođe može direktno pristupati memoriji – INC WORD PTR [EAX] će na primer uvećati za jeden reč koja je upisana u ćelije na koje ukazuje EAX.

Pri končenju ove instrukcije treba obratiti pažnju na vazan izuzetak ukoliko koristite registre ESP i EBP instrukcija neće delovati na podatak koji se nalazi u *data segment*-u nego na odgovarajuću adresu *stack segment*-a. Što se tiče ESP stvar je sasvim prirodna pošto se on bavi slekom. EBP je ovako „usmeren" zato što se pri prevođenju programa pisanih na višim jezicima koristi za adresiranje argumenata potprograma a ti argumenti se obično prenose baš preko steka.

Adresiranje baza plus ofset

Bazno adresiranje je veoma logično jer „gađa" celiju na koju ukazuje registar. Ponekad je međutim pogodno da registar ukazuje na početak neke fiksne strukture dok se u okviru same instrukcije navodi rastojanje od početka te strukture do željenog podatka. Pogledajmo na primer sliku 4 definisane je struktura KOORD koja ima tri komponente – X Y i Z koordinatu neke tačke. Zatim je za tu strukturu rezervisan prostor u memoriji (KOORD<>) efektivna adresa tog prostora upisana u ESI (LEA = *Load Effective Address*) i najzad njenim komponentama je pristupano sa MOV EAX [ESI] X odnosno [ESI] Y ili [ESI] Z.

Neka vas ne zavara što je pominju nekakve „promenjive" X Y i Z – radi se naprosto o simboličkim imenima pojedinih komponenti strukture pri čemu asembler već u trenutku prevođenja „zna" da je X=0 Y=4 i Z=8 pa sa ovi fiksni podaci ugrađuju u polje *displ* (u „intel"-ovoj terminologiji se oni zove *displacement*) odgovarajućih instrukcija. Ofsel je dakle, kod ovog adresiranja fiksan.

Adresiranje Indeks plus ofset

Situacije u kojima je ofsel fiksan su u praksi relativno retke – mnogo je češći slučaj da je početak nekog niza na fiksnom mestu i da on da treba da pristupimo nekom njegovom elementu čija se adresa menja u petlji. Brojac le petlje će, naravno biti neki od registara u praksi se često koristi ESI. Pogledajmo na primer instrukciju MOV AH 2A50h[ESI] – po neiasku ne nju mikroprocesor uzima sadržaj registra ESI, uvecava ga za 2A50h (ovo sabiranje se prirodno obavlja u nekom internom registru 80486 – sadrzaj ESI ce po izvršenju instrukcije biti očuvan) i tako dobija adresu memorijske celije u DS segmentu čiji će sadržaj preneti u AH. Svi registri opšte namene **osim** ESP mogu da se koriste kao Indeksni.

U praksi se, naravno, vrlo retko navode konkretne adrese – umesio toga rezervisacemo prostor za niz VEKTOR sa VEKTOR DD 100 DUP(?) a onda mu pristupali sa MOV EAX VEKTOR[ESI] ili SUB VEKTOR[EAX] 8.

Prema onome što smo do sada napisali rekli bi se da se adresiranje „Indeks plus ofset" može svesti na „baza plus ofsel" – sadržaj registra se i u jednom i u drugom slučaju sabira sa nekom konstantom. Pa ipak postoji jedna suptilna razlika koju ćemo upoznati studirajući C program sa slike 5 i dve njegova (moguća) prevoda na mašinski jezik.

Program sa slike 5 računa zbir kvedrata niza (recimo 32 bitnih) brojeva. U radnoj petlji se zapravo menjaju dve vrednosti – indeks *i* i ofsel elementa *v[i]* – kada je na primer i=5 ofsel bi trebalo da bude 5*4=20 pošto svaki elemani zauzima po četiri bajta. Na nekom primitivnijem mikroprocesoru to bi se moralo svesti na množenje kao u prvom prevodu sa slike 5. Drugi prevod međutim koristi jednu specifičnost mikroprocesora 80386/80486 – MOV EAX V[ECX*4] „množi" sadržaj ECX sa 4 i direktno pristupa potrebnom bajtu memorije a opet se ta instrukcija izvrši za **isto** vreme kao MOV EAX V[ECX] što znači da drugi prevod obezbeduje daleko racionalniji rad. Ovo „množenje" se u „intel"-ovoj terminologiji zove skaliranje (*scaling*) i ograničeno je na vrednosti 1 2 4 i 8 pri čemu. jesno množenje sa 1 nema mnogo smisla.

Videli smo da se za indeksirano adresiranje ne može koristiti ESP dok EBP i ovde predstavlja izveslan izuzetak – dok je kod baznog adresiranja deloveo isključivo na stek segment sada se pod odgovarajucim uslovima odnosi i na DS (slika 6). Ako je u instrukciji naveden bazni i indeks registar i ako je jedan od njih EBP smatra se da je EBP bazni registar ukoliko nije naveden i skalirajuci faktor

Nastavlja se na strani 80

УНЕСИТЕ

ПРОМЕНЕ

У СВОЈ НАЧИН ПОСЛОВАЊА

AB soft radi za Vas

KNJIGOVODSTVENI SOFTVERSKI PAKETI: FIPO (FINANSIJSKO KNJIGOVODSTVO)
LIDO (LIČNI DOHOCI) • ROK (ROBNO KNJIGOVODSTVO) • MAT (MATERIJALNO KNJIGOVODSTVO)
MAGIC (MAGACINSKO POSLOVANJE) • OSA (EVIDENCIJA OSNOVNIH SREDSTAVA) • VIR (VIRMANI)
BLAD (BLAGAJNA) • GRACE (EVIDENCIJA ČEKOVA) • KOM (POSLOVI KOMERCIJALE) • OKAM (OBRAČUN KAMATA)

CAR (PRODAJA VOZILA) • TEP (TEHNIČKI PREGLED VOZILA) • EVA (EVIDENCIJA AMBALAŽE)
MAP (MALOPRODAJA) • TAG (POSLOVANJE TURISTIČKIH AGENCIJA)

RAČUNARSKA OPREMA, ŠTAMPAČI, MREŽE...

AB SOFT, BEOGRAD, Kneza Miloša 82, tel/fax 656-857, tel 644-255 lok. 113

PC-AT 386/SX 33, HDD 105MB, FDD 3.5", 2MB RAM, SVGA COLOR 1950 DEM.

PC-AT 386/DX.40, 128 CASH, HDD 120MB, FDD 3.5", 5 25", SVGA COLOR 2550 DEM.

PC-AT 486/DX 33, 256 CASH, HDD 215MB, FDD 3.5", 5.25", SVGA COLOR 3500 DEM.

SVE VRSTE KOMPJUTERSKIH PERIFERIJAI

POSEBNI POPUSTI ZA DALJU PRODAJU!

DIHO COMPUTER SYSTEMS
BEOGRADSKA 41
TEL. 011-341-392, 496-351

ADACOM

286 / 20 MHz	1 MB	1,450
386 sx / 25 MHz	2 MB	1,550
386 / 40 MHz	4 MB	1,990
486 / 33 MHz	4 MB	2,900
486 / 50 MHz	4 MB	3,400

fax/modem interni DISCOVERY	400,-
fax/modem externi DISCOVERY	600,-
modem interni DISCOVERY	350,-
modem externi DISCOVERY	400,-
Grafička kartica ATI ULTRA	900,-
Grafička kartica S3 Windows	550,-
Joystick MS-500	50,-
koprocesor 287-20 MHz	100,-
koprocesor 387-20 MHz INTEL	250,-
koprocesor 387-40 MHz IIT	250,-
Miš GM-6	50,-
Miš GM-6000	80,-
memorija za IIIP/IIIP	350,-
Mobile rack za 3½" hard disk	160,-
Monitor filter staklem	65,-
Mouse pad	10,-
traka za LX-400	15,-
traka za LQ-550	15,-
traka za FX-1050	15,-
toner za IIP/IIIP	250,-
štampač HP IIIP bez tonera	2900,-
štampač EPSON LX-400	550,-
štampač EPSON FX-1050	1350,-
UPS 550 VA	600,

STANDARD ZA KONFIGURACIJU

- hard disk CONNER 80 MB 17 ms
- floppy disk PANASONIC 5¼"
- combi controller (2S, 2P, 1G)
- monohromatski VGA monitor
- TRIDENT 9000 512k
- AT desktop kućište
- tastatura 101 taster

DOPLATA ZA	
1 MB memory SIMM/SIPP	75,-
TRIDENT SVGA 8900 1 MB	50,-
OAK 087 1 MB	75,-
TSENG ET4000 VESA 1 MB	100,-
TSENG ET4000 enhanced 32k boja	180,-
S3 windows akcelerator	400,-
ATI ULTRA sa koprocesorom i milem	750,-
floppy disk 3½ 1.44 MB	130,-
hard disk CONNER 120 MB 19 ms	130,-
hard disk CONNER 170 MB 16 ms	260,-
hard disk CONNER 210 MB 13 ms	550,
GALAXY IDE cache controller	550,-
mini TOWER 200 W	30,-
midi TOWER 200 W	70,
slim line	70,
color monitor TOPFLY 1024x768	450,-
PHILIPS BRILLIANCE 7CM3209	700,-

Tel: 629-233 Fax: 629-672 BEOGRAD, Čika Ljubina 12 PRO MUSICA

ELEMENTARNE METODE

Seriju tekstova o algoritmima nastavljamo operacijom koja je najuže povezana sa programerskim problemima – pretraživanjem. Naravno, polazimo od najprostijih metoda, a bavićemo se i nekim pojmovima koje smo već pominjali.

Bojan Petrović

Normalno pretraživanje zamišljamo kao proces pronalaženja traženа informacije u skupu datih informacija, a na osnovu datog ključa iz nekog skupa ključeva. Recimo pri pretraživanju rečnika ključevi su reči a informacija podaci koji se dobijaju (sinonimi izgovor, značenje). U telefonskom imeniku ključ može biti prezime a informacija telefonski broj i adresa (ili obrnuto?). Pošto se obično očekuje veća količina informacija srodnih datom ključu, ona se grupišu u zapise a zatim se posebnim rešenjem iz njih dobijaju traženi podaci.

Aplikacija koje se bave pretraživanjem su veoma rasprostranjene: banka mora da prati sve račune svojih klijenata i stalno pretražujući podatke, kontroliše nastale transakcije. Sistem za prodaju avionskih karata funkcioniše na sličan način, uz mnoge specifične mogućnosti. Ustvari, za savladavanje raznih metoda pretraživanja najbolji je već pomenuti primer rečnika – za program, tabela simbola je rečnik imena objekata su ključevi, a zapisi sadrže informacije vezane za imenovani objekt.

Kao i kod redova prioriteta, zgodno je algoritme pretraživanja zamišljati kao pakete koji implementiraju razne operacije nad datim strukturama, a koje se mogu upotrebiti i odvojeno od matične grupe. Neke operacije od interesa su:

- inicijalizacija rečnika
- pretraživanje sa datim ključem
- umetanje novog zapisa
- brisanje određenog zapisa
- spajanje dva rečnika u jedan
- sortiranje rečnika

Listing 1

```
struct node
{ int kljuc, int info  };

struct node a[MAXN+1],
int N,

void seq_init(void)
{ N=0; },

int seq_search ( int v )
{
  int x=N+1,
  a[0] kljuc=v,
  a[0] info= 1,
  while (v!=a[  x] kljuc),
  return a[x] info,
}

void seq_insert ( int v, int info )
{
  a[++N]_kljuc=v,
  a[N]_info=info,
}
```

Pretraživanje pomoću niza

Listing 2

```
struct node
{ int kljuc, info, struct node *delje, },

struct node *glave, *z,

void list_init(void)
{
  glave=(struct node *) malloc(sizeof *glave),
  z=(struct node *) malloc(sizeof *z),
  glave >delje=z,
  z >delje=z,
  z >info= 1,
}

int list_search ( int v )
{
  struct node *t = glave,
  z >kljuc=v,
  while (v>t >kljuc) t=t >delje,
  if (v!=t >kljuc) return z >info,
  return t >info,
}

void list_insert ( int v, int info )
{
  struct node *x, *t = glave,
  z >kljuc=v,
  while (v>t >delje >kljuc) t=t >delje,
  x=(struct node *) malloc(sizeof *x),
  x->delje=t >delje,
  t >delje=x,
  x->kljuc=v,
  x >info=info,
}
```

Pretraživsnje pomoću povezanih lisli

Listing 3

```
int bin_search( int v )
{
  int l=1, int r=N, int x,
  while (r>=l)
  {
    x=(l+r)/2,
    if (v<a[x]_kljuc)
      r=x-1,
    else
      l=x+1,
    if (v==a[x] kljuc)
      return a[x]_info,
  }
  return  1,
}
```

Binarno pretraživanje

Listing 4

```
struct node
{ int kljuc, info, struct node *l, *r, },

struct node *t, *glave, *z,

void stabla_init(void)
{
  z=(struct node *) malloc(sizeof *z),
  z >l=z,
  z >r=z,
  z >info= 1,
  glave=(struct node *) malloc(sizeof *glave),
  glave >r=z,
  glave >kljuc=0,
}

void stabla_insert( int v, int info )
{
  struct node *p, *x,
  p=glave,
  x=glave >r,
  while (x!=z)
    { p=x, x=(v<x >kljuc) ? x >l   : x >r, }
  x=(struct node *) malloc(sizeof *x),
  x >kljuc=v,
  x >info=info,
  x >l=z,
  x >r=z,
  if (v<p >kljuc) p >l=x; else p >r=x,
}

int stabla_search ( int v )
{
  struct node *x = glave >r,
  z >kljuc=v,
  while (v!=x >kljuc)
    x=(v<x >kljuc) ? x >l   : x >r,
  return x >info,
}
```

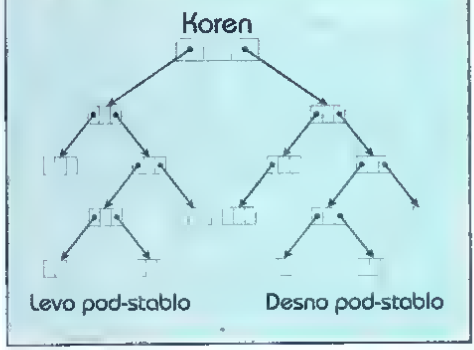
Pretaživanje pomoću binarnih stabala

Listing 5

```
void stablo_delete ( int v )
{
  struct node *c, *p, *x,
  x->kljuc=v,
  p=gleva,
  x=gleva >r,
  while (v!=x >kljuc)
    { p=x, x=jv<x >kljuc) ? x >l : x->r, }
  t=x,
  if (t >r==z) x=x >l,
  else
    if (t >r->l==z)
      { x=x->r, x >l=t >l, }
    else
      {
        c=x->r,
        while (c >l >l != z) c=c >l,
        x=c >l,
        c >l=x >r,
        x >l=t >l,
        x >r=t->r,
      }
  free(t),
  if (v<p >kljuc) p >l=x, else p >r=x,
}
```

Brisanje zapisa u binarnom stablu

Slika 1



Binarno stablo

U pojedinim slučajevima, pokušacemo da neke operacije spojimo u jednu, pošto tako može biti zgodnije recimo fuzija umetanja i sortiranja može biti od velike pomoći kod pojave duplikata ključeva

Ovu pojavu možemo obradit' na više načina Pre svega možemo jednostavno insistirati da naš rečnik sadrži samo zapise sa različitim ključevima a da ti zapisi sadrže i pokazivače na povezanu listu zapisa sa odgovarajućim ključem Ovde, dakle, jednim prelraživanjem indirektno dobijamo sve zapise sa datim ključem Drugi način je da sve duplikalne zapise ostavimo u rečniku i vraćimo prvi na koji naletimo Sada moramo obezbediti dodatni algoritam za pronalaženje preostalih što je sve zajedno dobro za aplikacije koje rade sa pojedinačnim zapisima, ali loše sa programerskog stanovišta Treca mogućnost je uvođenje posebnih identifikatora za svaki zapis na osnovu koga ćemo u skupu zapisa se istim ključem naći odgovarajući I tako dalje Mi ćemo nadalje podrazumevati mogućnost pojave zapisa sa istim ključevima i nečemo tome eksplicitno posvetiti pažnju

SEKVENCIJALNO

Najjednostavniji metod je smeštanje zapisa u niz Novi zapisi se dodaju na kraj niza a pretraživanje vršimo sekvencijalno (redom) Implementacija na listingu jedan podrazumeva ključeve („kljuc") i informacije („info") lipa celobrojnih veličine Mada će stvarne potrebe sasvim sigurno zahtevati komplikovanije zapise i ključeve, to nece imati nekog uticaja na suštinu stvari (promena **int kljuc** u **char *kljuc**, vodi do promene u **while** izrazu **while (strcmp(v,a[--x].kljuc))** čime smo uveli strangove)

U dalo j implementaciji koristili smo jednostavan način da obezbedimo da će se pretraživanje uvek završiti a da pri tom koristimo samo jedno ispitivanje Umesto da u jednom trenutku proveravamo da li je **x==0** a u drugom da li je **a[--x]** traženi ključ spojili smo ih u jedno u **a[0]** smo postavili traženi ključ Ovo je analogija sa našim ranijim slučajevima kada smo kao ograničenje koristili najmanji ili najveći ključ Istovremeno eliminišući jedno ispitivanje smanjili smo unutrašnju petlju što po ranijim razmatranjima, drastično može ubrzati rad algoritma

Sekvencijalno pretraživanje koničenjem niza „proši" N+1 poređenje za neuspešno i u proseku N/2 poređenja za uspešno pretraživanje U prvom slučaju zaključak sledi direktno iz koda svaki zapis mora biti obraden da bi se ustanovilo nije li naš ključ prisutan (ovo nije čudno objašnjenje – šta ako je niz sortiran? da li tada moramo svaki zapis proveriti?) U drugom slučaju podrazumevamo da je podjednako verovatno da će se zapis naći na bilo kom mestu u nizu pa sledi

$$\frac{1+2+3+\cdots+n}{N}=\frac{\frac{N+1}{2}}{N}$$

što je u skladu sa gornjim

Prirodno unapređenje sekvencijalnog pretraživanja pomoću niza bi bilo uvođenje povezanih lisli Naravno, ne zato što je se njima lakše operisati vec zato što je pomocu njih lako održati niz sortiranim Jedna moguca implementacija data je listingom dva Pošto je lista sortirana osnovni efekat je kao što je vec naglašeno mogućnost ranog prekida pretraživanja Onog trenutka kada se isposlavi da naš ključ nije manji od ispitivanog pretraživanje obustavljemo Sličan efekat smo mogli u prethodnoj implementaciji postići sortiranjem niza posle svakog umetanja što u nekim aplikacijama može biti konsnije Opet, pretraživanje ograničavamo uobičajenim postupkom za povezane liste – korišćenjem krajnjeg noda (obično se inicijalizuje na NIL, mada smo mi koristili nešto čudniji metod)

Sekvencijalno pretraživanje koničenjem povezanih lisli „proši" oko N/2 poređenja i u neuspešnom i uspešnom pretraživanju U prvom slučaju sa zakljućak izvodi kao i prethodni pot, dok je u drugom nešto drugačije polazi se od pretpostavke da je verovatnoca završetka rada jednaka na krajnjem čvoru i bilo kom od prethodnih

Naravno postoje i drugi metodi razmeštanja zapisa u lisli Verovatno najefikasniji je da se raspored vrši prema relativnoj frekvenciji pristupe pojedinim zapisima Optimalno je dakle smestiti na početak one zapise koji se najčešce traže Ova tehnika međutim poslije stvarne efekte lek ako je skup lih najčešce traženih zapisa relativno mali Ako informacija o frekvenciji pristupa nije dostupna, možemo se poslužiti metodom „samoorganizacije" niza Kad god pristupimo nekom zapisu (dakle kada smo ga „pronašli") pomerimo ga na pocelak liste Ova metoda se mnogo lakše iz očiglednih razloga implementira sa povezanim listama

BINARNO

Jedna od metoda koju smo ranije konstili je bila „podeli-pobedi" („devide-and-conquer") podeli skup zapise u dva dela odredi kom skupu pripada ključ i nastavi sa njim na isti način Binarno pretraživanje je jedna manifestacija ovakvog načina razmišljanja a primenjuje se kod većih količina podataka

Osmotrimo listing III Da bismo našli ključ „v" uporedimo ga sa onim na srednjoj poziciji u nizu Ako je manji učinimo isto sa levom polovinom niza ako je veći sa desnom u protivnom smo ga našli Dakle iz ovog sledi dva (očigledna) zaključka niz mora biti sortiran a metoda je prirodno rekurzivne Kao i kod QUICK i RADIX EXCHANGE sortova koristimo pokazivače „r" i „l" da ograničimo deo niza koji obrađujemo Kada taj podniz postane prazan zaključujemo da je pretraživanje bilo neuspešno U suprotnom ili će levi pokazivač postati **x+1** (gde je **x** sredina podniza) ili će desni postati **x-1** ili će **a[x]** biti baš traženi ključ

Održavanje niza srtiranim moze biti bolan posao Recimo ubacivanje novog najmanjeg elementa u niz povlači pomeranje svih ostalih za jedno mesto U proseku ova metoda troši N/2 pomeraja u proseku što je skupo To znači da je ova meloda generalno bolja za one slučajeve kada nam je del vec sortirani niz kada binarno pretraživanje stvarno sija

Ako se dogodi da postoji više zapisa sa istim ključem pretraživanje će se završiti negde u bloku sačinjenom od lih zapisa (naravno ne baš obavezno u sredini) Ako su nam ko zna zbog čega potrebni svi moračemo da dodelno skeniramo sekvencijalno niz levo i desno od onog pronađenog Vrlo slično se onda rešava i problem nalaženja zapise čiji ključevi upadaju u neki interval

Jedno moguce poboljšanje binarnog pretraživanja je unapređivanje načina pogađanja intervala u kom se traženi ključ (valjda) nalazi Ideja dolazi od prirodnog načina razmišljanja kada u telefonskom imeniku trazimo ime koje počinje sa B trazimo pri početku a ne sredini Ovaj metod interpolacije zahteva jednu relativ

no trivijalnu modifikaciju algoritma umesto r-njeg

x=(l+r)/2 (što je proizašlo iz **x=l+(r-l)/2**)

uvodimo

x=l+(v-a[l] key)*(r-l)/(a[r] key-a[l] key)

što daje deleko bolji (čitaj „približniji") pogodak. Performanse koje proizilaze iz ovog se lako mogu proveriti. Izražen o brojevima, klasično binarno pretrazivanje vrši ne više od lgN+1 poređenja u bilo kom slučaju, dok poboljšano binarno pretraživanje konsti ne više od igigN+1 poređenje. To otprilike znaci da će ova druga metoda svaki ključ pronaći u svega nekoliko poređenja, i što je podjednako bitno vreme rada ove modifikacije raste veoma sporo sa brojem elemenata.

Konačno, kod naše modifikacije par reči upozorenja. Red metode zavisi od raspodele ključeva po datom intervalu tako da se u praksi mogu desiti iznenađenja. Također, pošlo je uvođenio dodatno računanje za malo N razlike izmedu dve metode nisu dovoljno velike da bi se interpolacija isplatile. Tek pri dovoljno velikom broju elemenate poboljšana varijanta binarnog pretraživanja počinje da deluje.

BINARNIM STABLOM

O binarnim stablima je već bilo reči u ovoj seriji tekstova, tako da se nećemo zadržavati na njihovom opisivanju. Prisetimo se samo da je osnovna odlika stabla činjenica da svaki čvor stabla ima dva ili manje sinova (leve i desne veza) i jednog i samo jednog oca, sem korena stable koje ima samo sinove. Binarna stabla se odlikuju dodetnom osobinom da su svi zapisi levo od datog manji od njega a oni desno veći. Ovakva struktura savršeno odgovara binarnom pretraživanju.

Da bismo našli zapis sa datim ključem poredimo ga sa onim u korenu stable. Ako je manji idemo levim podstablom, eko je veći desnim, inače stanimo. Proces se zatim ponavlja rekurzivno na dva podstabla (ili manje). U svakom trenutku zahvaljujući uslovu koji se poštuje pri formiranju stabla (element manji od datog ide levo, veći desno), garantovano je da će se traženi ključ nalaziti u trenutnom podstablu (eko ga u stablu uopšte ima) i nigde drugde. Procedure se zaustavlja u trenutku kada smo ključ našli, odnosno kada više nema podstabala koje nismo obradili. Upozorimo de je razlike između binernog pretraživanja i pretraživanja binarnim stablom velika, te da se mora dobro razumeti.

Pri formiranju binarnog stabla, zgodno je da uvedemo dodatni čvor koji će imati samo jednog sina i to baš koren stable. U ovaj čvor ćemo staviti neku zgodnu veličinu, recimo najmanju od skupa vrednosti koje očekujemo (recimo nule?). Kao i kod sekvencijalnog pretraživanja u završni čvor postavljamo traženi ključ, iz potpuno istih razloga. Na taj način njiedno stablo neće stvarno postati prazno, a svako pretraživanje će biti „uspešno" završni čvor će nam ipak vratiti umesto stvarne informacije neki indikator (minus jedan?).

Prazno stablo će, na osnovu gornjeg sadržati čvor „glava" koji će pokazivati na završni. Implementacija rada sa ovakvim stablom data je listingom ćetiri. Završni čvor, kao i ranije, pokazuje na sebe. Da bismo u binarno stablo umetnuli zapis, vršimo neuspešnu pretegu za njim i na mesto završnog čvora do kog smo stigli postavljamo novi zapis (ključ). U implementaciji na datom listingu, umetanje čvora sa ključem koji već postoji u stablu se vrši desno od njega. To istovremeno znači da potrage za svim zapisima sa istim ključem podrazumeva nalaženje samo onog prvog, dok su ostali poredani jedan za drugim (jedna od mogućih varijanti).

Očigledno, sortiranje sa ovom strukturom dobijamo praktično besplatno, jer je kao takvo podržano definicijom strukture. Performanse pretraživanja zavise, zamislite od oblika stable (slobodno rečeno). U najboljem slučaju, čvorovi će biti simetrično raspoređeni oko osovine stabla (koja polazi od korena), a između korena i svakog eksternog čvora biće lgN čvorova. U proseku, možemo očekivati logaritamski red složenosti algoritma za pretraživanje. Prvi element koji se ubacuje u stablo postaje njegov koren. Sledećih N elemenata, generalno gledano, biće podeljeni ovim elementom na dva podskupa jednake veličine (što je idealan slučaj). Malo bolje ispitivanje će nam reći da je u pretrazivanju potrebno 2lnN poređenje pri korišćenju binarnog slable.

BRISANJE

Meda su sve operacije obrađene gore lake za razumevanje i jednostavne za implementaciju (rekurzivno), ostalo nam je brisanje čvorova stabla. Ova procedura može biti krupan zeolagaj i pored zgodne strukture stabala i mogućnosti rekurzije, te smo zbog toga njoj posvetili poseban odeljak.

Brisanje čvora je naravno, lako ako on nema dece – prosto odgovarajuću vezu njegovog oca prepravimo. Čak i ako postoji samo jedno dete, poseo nije težak – umesto datog čvora postavimo to jedno dete. Problem nastaje kod one preostale grupe čvorova, gde svaki ima po dva sina.

Jedna od metoda je zamena čvora koji brišemo sledećim po veličini. Očigledno je da će takav čvor imati najviše jednog sina, pa ga je lako uklonili (prema ranije rečenom). Procedura data listingom pet briše iz binarnog stable prvi čvor se datim ključem. Variable „p" čuva podatak o ocu čvora „x". Program prvo postavlja vrednost „p" (tražeći mesto „t") da bi po brisanju čvora „t" mogli de vršimo povezivanje. Zatim se proveravaju mogući slučajevi: ako „t" nema samo desno dete, onda će posle brisanja dete čvora „p" postati levo dete čvora „t"; ako čvor „t" nema samo levo dete, po brisanju će desno dete postati dete čvora „p" uz kopiranje leve veze čvora „t". Ako postoje oba sina „x" postaje najmanji čvor u desnom podstablu čvora „t" a desna veza tog čvora se kopira na levo mesto svog oca.

Postoje i razne druge metode za brisanje čvorova u binarnom stablu, mada je gornja uobičajena. Konačno, za pretraživanje je uobičajeno da se najviše vremena potroši na dobro brisanje. U nekim slučajevima, može se primeniti i „veštačko" brisanje: kada čvor nekim markerom samo označimo kao obrisan, ostavljajući ga u strukturi. Veliki broj tako „brisanih" čvorova može voditi do preteranih gubitaka u prostoru ili vremenu, što se u realnim aplikacijama često ne događa. Naravno, stablo se može periodično čistiti od „brisanih" čvorova i time naći neki kompromis.

I DALJE

Meda su binarna stabla opšte prihvaćena i dobro obrađena, imaju svoje loše osobine. Pojava „najgoreg slučaja" je ono čega se programeri najviše plaše. Setimo se kod sortiranja, koliko problema se moglo stvoriti sortiranjem već uređenog niza, ili, još bolje, niza koji je sortiran u suprotnom smeru. Tada je, čak i kod tako dobrih metoda kao što je QUICK sort, vreme rada raslo daleko preko nekih mnogo lošijih algoritama (u srednjem slučaju). Kod binarnih stabala se ovo manifestuje kod uslove balansiranosti stable o čemu će biti reči u sledećem nastavku. Dakle, pred nama su balansirana stabla.

Nastavak sa strane 75

Ukoliko je skalirajući faktor naveden, smatra se da je EBP indeks-registar pa se referiše DS.

Mikroprocesor 8088 i 8086 su vrlo sporo obrađivali instrukcije koje koriste ovaj adresni mod – ponekad je bilo potrebno čak 17 ciklusa za samo izračunavanje adrese. Zato su programeri steki navikli da ovo adresiranje, ako je to ikako moguće izbegavaju. Više nema razloga za to – 80386 i 80486, zahvaljujući znatno bolje rešenoj fazi pripreme naredbi troše isti broj ciklusa, bez obzira na tip adresiranja. Uz jedan mali ali značajan izuzetak – 80386 troši jeden (ali samo jedan) dodatni ciklus pri obradi instrukcija u kojima figuriše bazni i indeks registar. 80486 donosi dalje poboljšanja, tako da će ovaj ciklus nekada biti potreban a nekada ne – što je prosečno složenije to je teže precizno proračunati broj mašinskih ciklusa potrebnih za izvršenje nekog programa! 80486 će, međutim, dodati jedan ciklus (koga ne 80386 nije bilo) eko je vrednost upisana u bazni registar neposredno pre korišćenje ovog adresiranja. Ukoliko pišete neki kompajler obratite pažnju da ove dve instrukcije odvojite nekom radnjom koje će vam kasnije biti korisna.

Baza plus indeks plus ofset

Poslednje i najkomplikovanije adresiranje koristi se za pristup složenim strukturama smeštenim na slek ili matricama čije se bazna adrese nalazi u nekom registru. Ne sliči 7 prikazano je nekoliko primera ovog adresiranja koje ćete u praksi jako retko koristiti – reklo bi se da sadrži „jednu dimenziju više" od našeg normalnog razmišljanja. Pe ipak neki kompajleri će uz njegovu pomoć rado racionalizovati rad se parametrima.

Poslednji primer sa slike 7 je na prvi pogled vrlo neobičan – reklo bi se da su navedena dva ofsetal. Redi se međutim samo o pogodnom načinu pisanja instrukcije koju će asembler ispravno prevesti – u polje *displ* odgovarajuće instrukcije biće upisan zbir PT+Y.

Adresiranje steka

Mikroprocesor 80486 koristi stek na sasvim uobičajen način: instrukcijama PUSH i POP na njega se odlažu a sa njega skidaju sadržaji pojedinih registara (ili ćek podaci iz memorije), sveki poziv potmograma (CALL) ili interapt smeštaju na stek odgovarajuće informacije neophodne za nastavak programa, dok registar ESP ukazuje ne poslednji podatak na steku – kada stigne novi ESP se umanjuje i podetek upisuje na dobijenu adresu. Ukoliko se podetak ne može smestiti na stek (vrednost ESP manje od duzine podatka), generiše se *stack fault* (INT 12).

Kao i ostele instrukcije, PUSH i POP na osnovu izabranog registra „znaju" koliko se bajtova smešta na stek. Preporučili bismo vam, međutim, da **uvek** na stek šaljete čitav 32-bitni podatak bez obzira što vam možda treba samo njegovih 16 ili 8 bita. Stvar je u tome što je 80486 optimizovan prema pretpostavci da se svaki podatak nalazi na adresi deljivoj sa četiri. Ukoliko na stek stavite neki podatak dug dva ili (ne daj Bože) jedan bajt ova sinhronizacija se gubi i sve buduće naredbe koje rade sa slekom izvršavaju se znatno sporije. S obzirom da većina programa intenzivno koristi stek jedan neoprezni PUSH će žestoko uticati na performanse čitavog programa, usporavajući ge za tridesetak procenata! Dakle, ček i ako ne koristite sve bitove obavezno ih pre slanja na slek popunite vodećim nulama (ili jedinicama, eko je broj negativan). Za to se često koriste instrukcije tipa MOVSX ili MOVZX.

"Malo je toga na svetu što neko ne može napraviti malo gore i naplatiti malo manje; oni koji razmatraju samo cenu, nužno su žrtve takvih ljudi."

John Ruskin (1810-1900)

MIMICO

Mi smo oni drugi. Bolji i skuplji.

MIMICO Grupa za informatički inženjering
Maršala Birjuzova 3, YU-11000 Beograd
Tel. (011) 182-848, Fax (011) 624-846

KNJIGA U KOJOJ SVE PIŠE

Bazu podataka čini skup nekih podataka: te podatke koriste korisnici baze, a ponekad i program koji upravlja bazom za smeštanje svojih internih informacija. Tu spadaju svi podaci koje nije potrebno imati stalno na raspolaganju: razne tabele, parametri i slično. U bazu se mogu smestiti i informacije o samoj bazi podataka: polja u datotekama, indeksi, relacije... Stvar odavno postoji i naziva se *rečnik podataka* – to je skup podataka o ostalim podacima u siatemu.

Nenad Batočanin

Prelazak sa nekog standardnog programskog jezika na *Clipper* obično je veoma uzbudljiv *Clipper* ima u sebi mnoge stvari za koje je potrebno satima pisati procedure na standardnim programskim jezicima. Međutim, on nije samo standardni jezik sa „ugrađenom" dodatnom bibliotekom funkcija, stvari je u konceptu. Jednostavno prisustvo mocnih funkcija dozvoljava programeru razmišljanje u „krupnijim" koracima. A takvim čizmama od sedam milja mnogo brže se stiže do postavljenog cilja.

Ovakvo „putovanje" se uvek može učiniti mnogo udobnijim nego što jeste. Prvi korak ka tome je izdvajanje složenijih funkcija koje se često javljaju u programu u posebne procedure. Ponekad je dobro napraviti sistem procedura koji obavlja određenu funkciju – time se neke standardne operacije daleko pojednostavljuju.

PROBLEMI SA INDEKSIMA

Relacionu bazu podataka u principu čini skup tabela. U *Clipper*u su te tabele memorišu u **DBF** datotekama. Za svaku **DBF** datoteku (tabelu) vezuje se jedna ili više indeksnih datoteka u kojima se čuvaju odgovarajući indeksi. Program mora sam da vodi računa o pridruživanju indeksnih datoteka odgovarajućim **DBF** datotekama – na žalost *Clipper* ju ne podržava nikakav automatizam. Iz ovoga proizlaze tazni problemi, koji drastično povecavaju vreme potrebno za svaku intervenciju na postojecim indeksima.

Kada je u programu potrebno otvoriti jednu tabelu (datoteku) mora se otvoriti **DBF** datoteka i sve podružene **NTX** indeksne datoteke. To se radi na tedbom **USE** na primer

```
USE Student INDEX Stu_Name,Stu_ID NEW
```

Dvim se otvara datoteka studenata i dve indeksne datoteke. U prvoj sa cuvaju indeksi po imenima a u drugoj po šiframa studenata. Elementarna pravila dobrog programiranja nalažu da se podaci o ovoj datoteci – njena struktura, opis svakog polja, indeksi, izrazi po kojima se vrši indeksiranje i slično – zapišu u nekom dokumentu koji prati program. Ovo je neophodno zbog kasnijeg lakseg snalazenja u programu. U suprotnom rizikujete da za mesec ili dva potpuno zaboravite znacenje pojedinih polja u datoteci a to vam svakako nece pomoci u razumevanju programa!

Svaka izmena **DBF** datoteke odražava se i na indekse koji su otvoreni *Clipper*to automatski radi i o tome nije potrebno voditi racuna. Zbog raznih neregularnih situacija (neslanak struje, rasipljivanje racunara...) može se desiti da indeksi ne budu dobro azurirani i da na odpovaraju podacima u osnovnoj datoteci. Tada je potrebno izvistiti komandu **REINDEX** koja iznova kraira sva otvorene indekse. To znači da bi na početku programa morala da postoji procedura koja ispituje indekse i ako se ustanovi da nisu korekini krairaju sa novi komandom **REINDEX**.

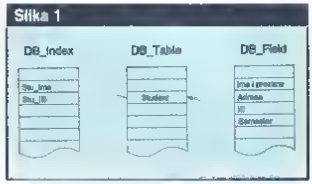
Na zalost, u nije kraj komplikacijama. Svaka indeksna datoteka na pocetku ima zaglavlje u koma se čuva izraz po koma treba izvršiti indeksiranje. Komanda **REINDEX** na osnovu ovog izraza kreira nove indekse. Ali šta ako je oštecano upravo to zaglavlje? Jedno je sigumo – u tom slučaju naredba **REINDEX** nece pomoci. Tada je neophodno kreirati novu indeksnu datoteku naredbom **INDEX ON**. Međutim ova naredba zahteva kao argument i izraz po kome se vrši indeksiranje. Iako da se mora pisati nešto popul ovog

```
USE Student NEW
INDEX ON Upper(s..name) TO Stu_Name
INDEX ON s_id TO Stu_ID
```

Datoteku studenata cele verovatno konstiti u više procedura. Ako se u tim procedurama menja datoteka **STUDENT**, menjaca sa i indeksi, pa je potrebno uvak drzati otvorene sve indeksa

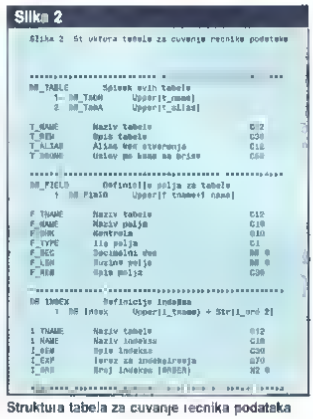
I tu bi priči (možda) bio kraj, da nije jednog važnog perioda u životu svakog programa: izmene. U iranuiku pisanja programa lako se može desiti da pogrešite u izboru indeksa ili vam kasnije zatreba još jedan indeks. Tada počinje posao koji možete praporučiti samo najgorim neprijateljima po celom programu. Treba pronaći sve naredbe **USE** i **INDEX**

Slika 1



Struktura recnika podataka

Slika 2



Struktura tabela za cuvanje recnika podataka

DN koje se odnose na tu datoteku i izvršiti odgovarajucu izmenu. Naravno, tu izmenu treba odmah uvrstiti i u vec pomenutu dokumentaciju programa. Na žalost, ovakve izmene mogu veoma lako da poremete i ostalak strukture programa. Na primer, ako je novi indeks ubačen na prvo mesto, onda naredba **SET ORDER TO** 1 više neće aktivirati indeks sa imenima, nego novi indeks.

Prikom instalacije programa potrebno je inicijalizovati kompletnu bazu podataka. To se uglavnom svodi na brisanje podataka iz svih datoteka ili na ponovno kreiranje svih potrebnih tabela. Za ovakvu operaciju neophodno je imati podatke o strukturi svih tabela koje ulaze u bazu podataka.

Znači, struktura **DBF** datoteka (uključujući i indekse) veoma utiče na program i dokumentaciju. Zato se pri izmeni strukture baze podataka (odnosno pojedinih **DBF** datoteka) dosta vremena troši na sve ove izmene. Treba izmeniti dokumentaciju, modul za kreiranje indeksa, modul za periodično brisanje i sve procedure koje koriste dotičnu datoteku.

U stvari, glavni problem je u zavisnosti programa i baze podataka: izmena u programu poviači nelku izmenu u strukturi baze podataka i obrnuto. Ovaj začarani krug je davno uočen i razrađene su tehnike koje ga donekle poništavaju. Osnovna ideja se svodi na **nezavisnost** programa i podataka. Nezavisnost se dobija uvođenjem nekoliko nivoa, od kojih svaki komunicira sa narednim samo strogo kontrolisanim kanalima i minimalno je zavisan od njega. Naravno potpuna nazavisnosi je samo teorijski pojam, ali se mnogo može učiniti teorijenizacijama programa koji upravlja bazom podataka.

Pošto je program toliko zavisan od baze podataka treba sve učiniti da se ta zavisnost smanji. Tako će i broj intervencija u slučaju izmene biti manji. Međutim, program **mora** biti donekle zavisan od strukture baze podataka. Na primer, da bi se ubrzalo pretraživanja program mora imati informaciju da li datoteka ima indeks. Zato se ova zavisnosi može smanjiti, ali se ne može potpuno izbeći.

REŠENJE: REČNIK PODATAKA

Pošto se svaka izmena indeksa mora raditi na nekoliko mesta, ima smisla za otvaranje svake datoteke praviti posebnu proceduru i tako promene indeksa vrsiti na samo jednom mestu. Međutim, to nas na spašava od izmene nekih procedura kao što je ona za kreiranje novih indeksa.

Ovde se moze primeniti jedno veoma jednostavno i elegantno rešenje: podaci o strukturi baze podataka (ukoliko se čuvaju u bazi podataka baš kao i ostali podaci). Taj deo baze podataka obično se naziva **recnik podataka**. Ova u suštini jednostavna ideja može veoma mnogo pomoći pri unutrašnjoj organizaciji programa.

Kod relacionih baza podataka dosta je jednostavno memorisati kompletnu strukturu baze podataka – u slučaju *Clippera*, svih **DBF** datoteka odgovarajucih polja, indeksa i relacija. Podaci o datotekama pojmu i indeksima mogu se čuvati u tri datoteke: **DB_Table**, **DB_Field** i **DB_Index**. Na slici 1 je prikazan međusobni odnos izmedu ovih datoteka. U svakom slogu datoteke **DB_Table** nalaze se podaci

Slika 3. Open

```
PROC Open (cFile, cAlias)
  LOCAL cName, aIndex, i, nLen

  // Spisak indeksa
  USE DB_Table INDEX DB_Idx, DB_Tabx NEW
  nLen  = len(i_ndex)

  // Traži tabelu
  SEEK Padr(Upper(cFile), nLen)

  IF !Found()
     Alert ("DATOTEKA " + Trim(cFile) + " NE POSTOJI U SPISKU ")
     RUET
  ELSE
     cName  = Upper(i_name)
     REPLACE t_alias WITH IIF (cAlias == NIL  cFile  cAlias)
  END IF

  // Otvori spisak indeksa u listoj zoni
  USE DB_Index INDEX DB_Indx
  DBSeek (cName +  0    )

  // Napuni niz indeksa
  aIndx  = {}
  WHILE Upper(i_tname) == cName
     AAdd (aIndex, Trim(i_ndex))
     SKIP
  END DO

  // Otvori tabelu i indekse
  DBUseArea ( F   , cName  cAlias)
  AEval (aIndex, { |i| DBSetIndex (i) } )

RETURN
```

Funkcija za otvaranje datoteke i svih indeksa

Slika 4 Order

```
PROC Creor (cIndex)
  LOCAL cName, aIndex, i  aSelect  cAlias

  // Zapamti oblast
  aSelect  = Select()
  cAlias  = Alias()

  // Spisak tabela
  USE DB_Table INDEX DB_TabA NEW
  FFFK Padr(Upper(cAlias)  Len(i_alias))

  IF !Found()
     Alert ("NE POSTOJI EVIDENCIJA O TEKUĆOJ RADNOJ OBLASTI ")
     RUET
  ELSE
     cName  = Upper(t_ndex)
  END IF

  // Otvori spisak indeksa u listoj zoni
  USE DB_Index INDEX DB_Index
  DBSeek (cName +  0    )

  WHILE Upper(i_tname( == cName
     IF Upper(cIndex( == Upper(Trim(i_name))
        (dSelect) >(DBOrdNumber(DB_Index >i_ord))
        EXIT
     END IF
     SKIP
  END DO

  // Zatvori tabelu
  CLOSE
  &(Select)

RETURN
```

Procedura za promenu aktivnog indeksa

o jednoj labeli u bazi. Podaci u ovoj labeli mogu se zabraniti po potrebi, ali sigurno treba uključiti naziv labele (datoteke): opis sadržaja parametre potrebne pri otvaranju i slično. Za svaki slog datoteke **DB_Table** vezano je više slogova iz **DB_Field** koji sadrže odgovarajuća polja. Na sličan način se za jednu labelu vezuju i indeksne datoteke čiji nazivi se čuvaju u datoteci **DB_Index**.

Rečnik podataka ne mora da sadrži samo osnovne podatke. Tu je zgodno ubaciti i mnoge druge informacije koje kasnije mogu dobro doći prilikom održavanja sistema. Na slici 2 je data moguća struktura ove tri datoteke. Tu su, pored osnovnih informacija i opis elemenata izrazi za indeksiranje i sl. Mnogi od ovih podataka se neće koristiti direktno ali će biti dragoceni pri izradi dokumentacije. Izradom ovakvog rečnika centralizovali smo sve informacije o strukturi baze podataka sa sasvim malim izuzecima. Zneti ideja je da ubuduće svaku izmenu baze radimo samo na jednom mestu, u rečniku podataka.

RAD SA REČNIKOM

Najkritičnije mesto kod rada sa indeksima je naredba **USE** odnosno otvaranje **DBF** datoteke i svih datoteka sa indeksima. Na slici 3. je prikazana funkcija **Open** koja otvara zadatu datoteku i sve odgovarajuće indekse kao i naredba **USE**. Prvo se na osnovu zadatog naziva datoteke izazi naziv u labeli. Ako ne postoji labela u evidenciji izdaje se odgovarajuće obaveštenje i prekida se rad programa (mada ima smisla i pokušati sa nastavkom uz određeno upozorenje). Ako sa datoteka nalazi u evidenciji otvara se datoteka u novoj radnoj zoni i upisuje se odgovarajući alias u rečnik. Korišćenjem ovog podatka može se kasnije lako otkriti pokušaj otvaranja već otvorene datoteke što može biti veoma korisno kod velikih aplikacija sa dosta nezavisnih modula. Zatim se iz datoteke svih indeksa izdvajaju slogovi koji se odnose na zadatu datoteku i upisuju u niz. Na osnovu ovog niza se otvaraju svi indeksi koji pripadaju datoteci korišćenjem funkcije **DBSetIndex**. Naredbu **USE** u programu sada treba zameniti funkcijom **Open**. Na primer

Open ('Student/")

Sada je problem dodavanja ili oduzimanja indeksnih datoteka resen: izmene treba raditi samo u rečniku podataka i u datoteci **DB_Index** a funkcija **Open** će uvek otvarati sve indekse koje nađe na spisku. Sa ovako otvorenom datotekom radi se sasvim uobičajeno potpuno isto kao da je za otvaranje upotrebljena naredba **USE INDEX**.

Ponekad je potrebno datoteku otvoriti bez indeksa ili samo sa nekim određenim indeksom – najčešće zbog uštede na brzini otvorenih datoteka. Tada se umesto funkcije **Open** može koristiti obična naredba **USE** – pošto se indeksi ne otvaraju kasnije izmene u broju ili redosledu indeksa neće se odraziti na ovu datoteku.

Ovakve centralizacija otvara zaista lepe mogućnosti za neke inače dosta komplikovane operacije. Na primer, vrlo je jednostavno napraviti kontrolu pri stupa podacima u datoteci za svakog korisnika se može voditi evidencija o tome kojim labelama može pristupati (na primer, opsl u nekom delu rečnika podataka) i na početku funkcije **Open** se ispita dozvola za rad sa labelom. Neretko pri tom kasnije treba ispitati mogućnosti naslavka tekuće procedure jer se može desiti da i rad bez te datoteke bude nemoguć. Takođe se olaksavaju neke dodatne operacije koje je potrebno uraditi pre otvaranje datoteke (provera prislupa kod rada u mrazi i sl.)

Ovakvo rešenje krije u sebi i neke nedostatke. Naime redosled otvorenih indeksa je u programu nevidljiv. Ako datoteku otvorite sa

USE Test INDEX Test1, Test2 NEW

naredba **SET ORDER TO 2** imaće sasvim vidljivo dejstvo (ili se barem lako čini!) dok se kod otvaranje sa **Open („Test")** to ne vidi u tekstu programa. Naravno u definiciji (u datoteci **DB_Index**) uvek se može videti koji je indeks definisan kao drugi ali programer to ne vidi u tekstu programe. Tako se može javiti sledeća greška: korisnik u rečniku doda još jedan indeks i postavi ga na mesto broj 2 tako da stari indeks broj 2 postaje treći. Time naredba **SET ORDER TO 2** u programu postaje pogrešna sada treba da glasi **SET ORDER TO 3**.

Problem se elegantno može rešiti uvođenjem funkcije sa slike 4. One menja aktivni indeks ali ne na osnovu rednog broja već na osnovu njegovog naziva. Na primer umesto **SET ORDER TO 2** koristi se funkcija **Order („Test2")** što će uvek postavljati indeks **Test2** za aktivan bez obzira na izmene u broju i redosledu indeksa.

Vrlo zgodne osobine opisanog sistema je to što na zahteva velike izmene u već napisanim programima: posle kreiranja rečnika podataka svi stari programi će savršeno korektno raditi bez obzira što sistemu koristi naredbu **USE** a ne funkciju **Open**. Novi sistem tako možete sasvim postepeno uključivati u svoje programe.

ODRŽAVANJE BAZE

Rečnik podataka je dragocen za kontrolu podataka u bazi. Najjednostavniji primer je kreiranje novih indeksa. U prethodnom delu je bilo govora o potrebi kreiranja novih indeksa, najčešće prilikom različnih neregularnih situacija. Ako podatke o indeksima svih datoteka imate na raspolaganju, problem je gotovo trivijalan jednostavno treba redom otvarati datoteke sa spiska i kreirati odgovarajuće indekse. Procedura **CreateIndex** koja to radi prikazana je na slici 5. Ovim se zaobilazi naredba **REINDEX** i problem sa oslećenjem indeksnih datoteka sada se sve kreira iz početka.

Podaci se u bazu podataka neprestano upisuju i mada se izvesni podaci i brisu rezultat je uvek pozitivan: baza podataka neprestano raste. Zbog ograničenog prostora na disku i brže obrade podataka potrebno je povremeno brisati stare podatke iz baze. Kod raznih poslovnih primena podaci se obično brišu posle periodičnih obračuna ili na kraju godine. Ovde se javlja problem sličan kreiranju indeksa pošto se struktura baze podataka menja uvek treba predvideti i brisanje novih labela. Stalne izmene procedura za povremeno brisanje baze je zaista mučan posao. Međutim, to se sada lako može izbeći korišćenjem opisa u rečniku za svaku labelu koju je potrebno periodično brisati navodi se uslov koji treba da bude zadovoljen da bi se slog datoteke proglasio starim i izbrisao. Ne primer uslov može biti "t_dat <= Date()" – ovim se iz datoteke brišu svi slogovi stariji od današnjeg datuma. Sada je dovoljno redom otvarati datoteke za koje je zadat uslov (polje t_dcond u **DB_Table**) i izvršiti naredbu **DELETE ALL FOR &uslov** – tako jedan zamoran posao po

staje potpuno automatizovan i krajnje jednostavan. Naravno, sada nije potrebna nikakva intervencija ni kod izmene indeksa pošto se svi podaci čitaju iz rečnika podataka dovoljno je izmenu izvršiti samo tamo.

Još citav niz poteza postaje veoma pojedno stavljen na primer, kreiranje potpuno nove baze podataka prilikom instalacije je trivijalan posao kada postoje informacije u rečniku podataka. Naravno kada pravja instalacionu verziju programa, osim **EXE** programa morate isporučiti i tri datoteke koje čine rečnik podataka.

Ako je povećana verovatnoća oštećenja baza podataka može se lako napraviti program koji proverava strukturu svake datoteke posebno. Ovde je problem što se i rečnik nalazi u **DBF** datotekama, pa šta ako se on ošteti? Onda se (bar za korisnika) gubi mogućnost pristupa svim podacima u bazi. Rešenje ovog problema može biti pomalo neobično na osnovu podataka iz rečnika generiše se standardni *Clipper* program koji proverava strukturu generišne nove indekse itd. Automatsko generisanje (znači ne piše ga programer već neki pomoćni program) ovakvog programa je jednostavno. Prednost nad ručnim pisanjem ovakvog modula je velika, jer se uloga programera sada svodi samo na održavanje rečnika podataka, dok se svi ostali poslovi obavljaju automatski. Znači, ako iz bilo kog razloga smatrate da je **EXE** datoteka s gumija od **DBF** datoteka rečnika predviđite mogućnost da u finalnoj verziji programa generišete module za proveru strukture i kreiranje indeksa i sl. U veoma osetljivim slučajevima mogu se predvideti obe varijante: baza se može kreirati na osnovu **DBF** datoteka i na osnovu modula za kreiranje, pri čemu se modul upotrebljava kada je rečnik oštećen. Provera da li su podaci u rečniku „dobri" se može obaviti uvođenjem kontrolne sume za sve njegova podatke.

IZMENA REČNIKA

Pošto su sada svi problemi svedeni na manipulaciju rečnikom potrebno je razraditi tehniku za upis i azuriranje rečnika. Rečnik se može menjati iz bilo kog programa koji može da menja strukturu i sadržaj **DBF** datoteke, ali je mnogo bolje napraviti pomoćni program koji će raditi. Upis i brisanje datoteka iz rečnika na mora se uvek raditi „van" programa, kod nekih praktičnih problema često može zatrebati da se kreira nova ili izbriše postojeća datoteka za vreme izvršavanja programa. Naravno, sam posao kreiranja veoma je jednostavan, ali problem je što onda rečnik podataka neće imati potrebne informacije o novim datotekama. Zato treba prilikom kreiranja (konstrukcijom funkcije **DBCreate** ili na neki drugi način) podatke o novim datotekama upisivati i u rečnik podataka. Na slici 6 prikazane su funkcije za upis i brisanje datoteka iz rečnika podataka.

Funkcija **CreateFile** na osnovu naziva datoteke i strukture prvo kreira datoteku, a zatim podatke o njoj upisuje u rečnik podataka. Treći parametar ove funkcije je niz parova **{Izraz, Datoteka}** koji definiše sve indekse za datu datoteku. Na osnovu ovog niza, funkcija upisuje podatke u rečnik i kreira nove indeks se. Funkcija za brisanje **DeleteFile** nasto je jednostavnija, na osnovu naziva datoteke briše se sve datoteke i opis u rečniku.

Time je zavisnost programa od datoteka smanjena, ali što se tiče indekse, sada je dovoljno manjali indekse samo u rečniku, jer se sve informacije odatle automatski koriste u programu. Naravno, zavisnost još uvek postoji, ako izbrišete neki indeks naredba **SEEK** u programu nema smisla. Ipak, rezultat je i te kako pozitivan.

Ostala je još jedna veoma važna kanka: program najviše zavisi od polja u bazi. Program mora „znati" koja su polja, kog tipa i veličine u datotekama. U nekim slučajevima se stvari može automatizovati. Ako se sa datotekom rade samo standardne operacije, nije problem napraviti univerzalne procedure za upis, brisanje, unos i pretraživanja podataka. Međutim, obično se realni problemi ne uklapaju u ovaj šablon, pa takve univerzalne procedure nisu baš sasvim komforne za krajnjeg korsnika. U suštini *Clipper* ne razlikuje naročito memorijsku promenljivu i podatak iz sloga **DBF** datoteke – i jedno i drugo je nešto što on razmotrivo može koristiti u daljoj obradi. Jedina razlika je u načinu dobijanja tog podatka. Bilo bi moguće izdvojiti sloj programa koji su formalski bitna o otvaranju datoteka, uspostavljanju relacija itd., dok bi ostalak programe jednostavno uzimao dobijene podatke. Rečnik podataka može predstavljati sasvim solidnu osnovu za takav sistem. Tu tek treba očekivati promenu objektno orijentisanog programiranja – izgleda da će novi *Clipper* stići u pravo vreme!

Slika 6

```
PROC CreateFile (cFile, aStruc, aIndex)
   LOCAL aSav := 1

   aSav := Select()

   // Kreiranje datoteke
   DBCreate (cFile, aStruc)

   // Indeksiranje
   IF aIndex <> NIL
      USE (cFile) NEW
      FOR i := 1 TO Len(aIndex)
         INDEX ON &(aIndex[i 1]) TO  aIndex[i 2]
      NEXT
      CLOSE

      // Upis indeksa
      Open ( DD_Index* )
      FOR i := 1 TO Len(aIndex)
         APPEND BLANK
         REPLACE  i_exp WITH aIndex[i,1],
                  i_name WITH aIndex[i 2]
                  i_ord WITH i
                  i_tname WITH cFile
      NEXT
      CLOSE

   END IF

   // Upis u recnik
   Open ( DD_Table* )
   APPEND BLANK
   REPLACE t_name WITH cFile
   CLOSE

   // Upis polja
   Open ( DD Field )
   FOR i := 1 TO Len(aStruc)
      APPEND BLANK
      REPLACE  f_tname WITH cFile,
               f_name WITH aStruc[i 1]  ;
               f_chk WITH aStruc[i 5],
               f_type WITH aStruc[i 2], ;
               f_len WITH aStruc[i,3],
               f_dec WITH aStruc[i,4]
   NEXT
   CLOSE

   Select (nOsv)
RETURN

PROC DeleteFile (cFile)
   LOCAL nDel  cTmp  cOldName

   nDel := Select ()

   // Fizicko brisanje
   ERASE (cFile + ".DBF")
   ERASE (cFile + ".NTX")
   ERASE (cFile +  ".DBT")

   // Brisanje iz recnika
   Open ( DD_Table )
   GO@ Pos@(Upper(cFile)  Len(& name))

   IF !Found()
      Alert ("NATUTKA GO PRSTALU U APlikaU" )
      QUIT
   END IF

   cOldName := t_name
   Select DD Table
   DELETE
   PACK
   CLOSE

   Open ("DD Index" )
   DELETE ALL FOR Upper(cOldName) == Upper(i_tname)
   PACK
   CLOSE

   Open ("DD_Field")
   DELETE ALL FOR Upper(cOldName) == Upper(f_tname)
   PACK
   CLOSE

   Select (nDel)

RETURN
```

Kreiranje nove datoteke i njen upis u rečnik podataka

Zavisnost programa od polja može se donekle ublažiti neke podatke koji su vezani isključivo za **DBF** datoteke treba izdvojiti u rečnik. Tako bi se za svako polje u rečniku mogli upisivati podaci o unosu i izmeni (formal kontrole podrazumevane vrednosti prave prislupa pojedinih konsnika itd.)

Bilo bi veoma dobro spojiti i poslednju karika u lancu izmena podataka o strukturi u rečniku i strukture **DBF** datoteke. Ovakva izmena obično se tade uz pomoć programa *DBU* koji omogucava kreiranje i izmenu strukture. Međutim, *DBU* veoma loše rešava problem pri nekim složenijim izmenama: program ne radi dobro i što je najgore ponekad uni štava podatke u **DBF** datoteci. Tipičan primer je promena naziva ili tipa polja. Zato je dobro konstit pro ceduru sa slike 7 koja menja strukturu datoteke u skladu sa rečnikom podataka. Znači, kada je potrebno da napravite izmenu napravicete je u rečniku a procedura **ChangeSiruc** ku izmenu upisuje u datoteku. Možda deluje prilično čudno, ali upotreba naredbi **COPY TO** i **APPEND FROM** za izmenu strukture dalinitivno otpada pri prvoj izmeni tipa polja ove naredbe na tade kako se očekuje. Zato se izmena strukture more obaviti posebnom procedurom. Procedura **ChangeStruc** prvo kreira nizove **aSt1** i **aSt2** koji sadrze strukturu datoteke u rečniku i u datoteci. Sada je potrebno izmeniti strukturu datoteke. Iako da opis u rečniku bude važeći, ali pri tom treba sačuvati što je moguca više podataka. Data procedura upotrebuje svaka dva elementa nizova struktura. Ako se ustanovi da su elementi si izvišće se jednostavno kopiranje polja. Ako je u rečnik upisano polje koje ne postoji u datoteci, polje treba dodati u datoteku.

Ako je promenjen samo naziv starog polja, kopira sa sadržaj polja sa starim nazivom. Ali kako razlikovati polje promenjenog naziva i novo polje? Jer može se izmeniti naziv starog polja **IME** u **NAZIV** ali potpuno isle situacije je kada se izbrise polje **IME** a zatim doda polje **NAZIV**. Zato je neophodno na neki način voditi evidenciju o „poreklu" polja kako bi se razlikovala ova dva slučaja. *DBU* problem rešava tako što strukturu datoteke prvremeno učitava u poseban niz, pa se kasnije laj niz poredi se izmenjenim. U rečniku je primenjeno nešto drugačije rešenje: svaka promena naziva polja pranosi se i na polje **F_CHK** u koma se takode čuva naziv polja. Pri upisu nove datoteke ovo polje se ažurira automatski. Kada treba izmeniti strukturu menjaju se samo polja **F_NAME** i **F_TYPE** dok **F_CHK** ostaje nepromenjeno. Na osnovu njega se odlučuje stari naziv polja (ako je izmenjen).

Izmena dimenzija polja takode se rešava jednostavnim kopiranjem, na primer polje **NAZIV** se iz **CHARACTER 30** menja u **CHARACTER 25**. Najveći problem je izmena tipa polja. Mada deluje čudno do ove akcije veoma često dolazi, nekad zbog greške a nekad i namerno. Funkcija **ChengeStruc** pokušava da konvertuje jedan tip podatka u neki drugi, koristeći standardne funkcije. Naravno „inteligentiju" ove konverzije treba vrlo oprezno prihvatiti.

Rezultat analize je niz tipa {**P_OLD, P_NEW, ACTION**} **P_OLD** i **P_NEW** su nazivi novih i starih polja a **ACTION** je izraz kojim treba zamanih staro polje u novom polju. Pre upisa nove strukture svaka ko treba dozvoliti izmenu ovih akcija, moguce je da konsnik želi da uradi neku sasvim prostu operaciju koja nije predvidena. Zalim se na osnovu ovih akcija kreira nova datoteka i u nju kopira sadržaj stare.

Rečnik podataka ne mora sadržali samo podatke o polima i indeksima – ju se sasvim lepo mogu uklopiti i podaci o relacijama, pa i podaci o nekim memorijskim promenljivim koje se dobijaju iz podataka u bazi. Moglo bi se reći da je sve ovo samo početak jednog uopštavanja problema – sledeci korak bi mogao biti pravljenje rečnika funkcija – u njemu bi se nalazile funkcije umesto promenljivih, njihovi opisi, prava prislupa podacima i slično.

Ali to je već sledeća priča!

Slika 5 CreateIndex

```
PROC CreateIndex
   LOCAL aSelect

   aSelect  = Select()

   // Spisak tabela
   USE dd_table NEW

   INDEX ON Upper(t_name) TO DD_TAMB
   INDEX ON Upper(t_alias) TO DD_TabA

   // Indeksi
   USE DD Index NEW
   INDEX ON Upper(t_tname) + Str(t_ord 2) TO DD_Index

   SELECT DD_Table

   DO|LE | Eof()|
      cName  = Upper(t_name)

      IF ! |Trim(cName) >= "SD_TABLE   OR  Trim(cName) == "DD_INDEX

         // Ned|j spisaP IndePen
         SELECT DD_Index
         cNameP |cName +   0    T |

         // Otvori tmbele
         DBUseArea | T ,   cName  "ENG"|

         // Kreiranje IndePen
         WHILE Upper(DD_Index->i_tname) == cName
            INDEX ON e|DD_Index >i exp| TO |DD_Index >i_name|
            DBIF ALIAS DD_Index
         END DD

         CLOSE
      END IF

      SELECT DD_Table
      DBIF
   END DD

   CLOSE DD Index
   CLOSE DD Table

   Select (aSelect)
RETURN
```

Procedura za kreiranje novih indeksa

Slika 7 ChangeStruc

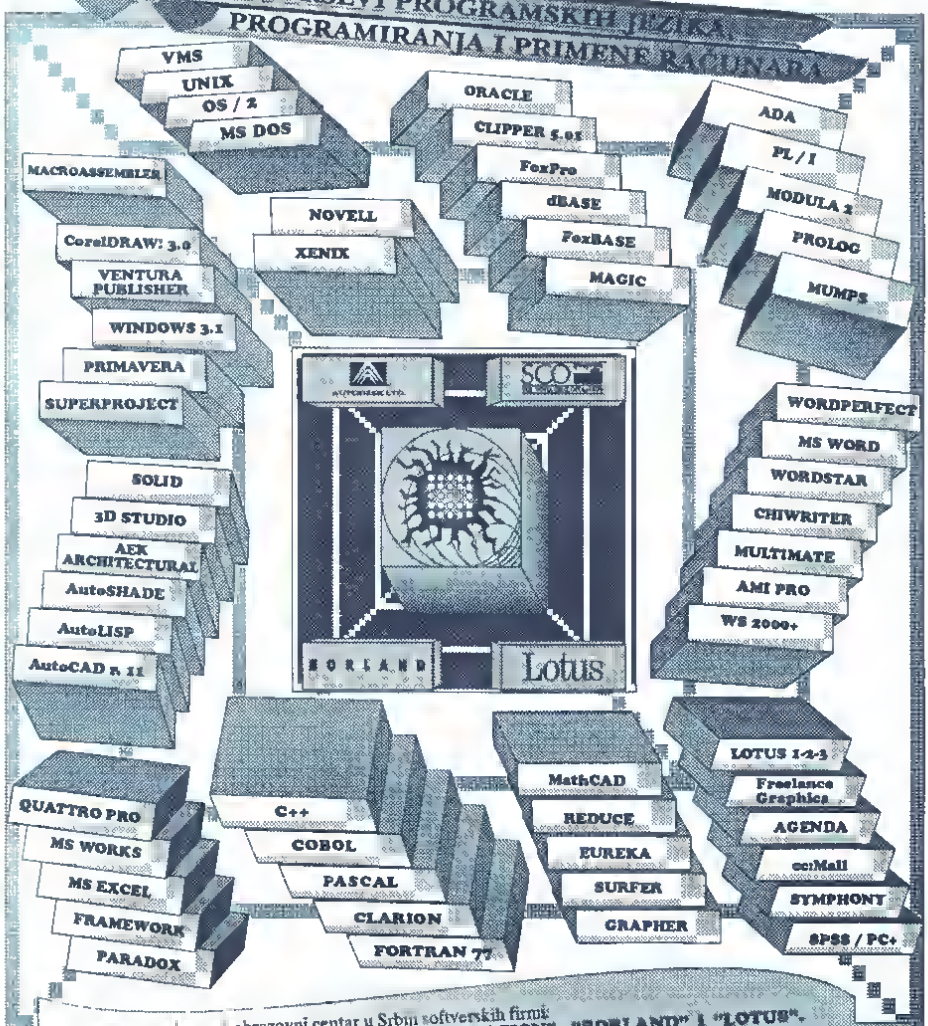
```
FUNC ChangeStruc |cFile|
   LOCAL aStr1  = {}, aStr2  = ||  t  nPos| n

   // Uzmi strukturu iz datoteke |stara struPlura|
   IF FIle |cFile + ' DBF |
      END |cFile|
      aStr1  = END|ruct||
      CLOSE
   ELSE
      aStr1  = {}
   END IF

   // Uzmi struPturu tm reonika |nNsa struktura)
   USE DB_Field
   GO TOP
   WHILE !Eof()|
      IF Trim|f_lname| == Trim|Upper|cFile||
         AAdd |aStr2, {f_name, Upper|t_type|, f_len  l_dec  t_cak}
      END IF
      SKIP
   END DO

   // Privremena tabela aKolje
   DBCreate |"__&TMP"  | ,
      {"@ICT"  "C , 12, 0|
      |"FILE"  "C   17  A|, ,
      |"ACTION", C , 40  0}
   ||

   USE __&TMP NEW

   // Paradjem|m 2 za strukturama 1  kreiranje pareva aku|je
   For t |= 1 TO |ovi@Str2|=
                                  ^v v
      // Kad|l odgovore|uou pozicija u strukturi 1
      nPos  = AScan |aStr1   | |s| Upper|Trim|s[1]|| == Upper|Trim|aStr2|i,1||
   } |

      // Pol|m nt|o novo
      IF nPos <> 0

         // Ako |o Le|l atribut, samo sa Poplrk
         IF Upper|aStr1|nPos 2|| >= Upper|aStr2|1 2|}

            // Ukvl 1 P| 1 PE
            APPUMN BLANK
            REPLACE Ftm WITH AS|aStr1|nPos||
                             Dlet WITH DD|aStr2|1|| |
                             Aciler WITH aStr1|nPos,3|

         // Atribut )m promen|es|
         ELSE

            // Svakako stavi t P| t P2
            APPE@D BLANK
            REPLACE Ftm  WITH AS|aStr1|nPos||   ;
                              Dlet D|TH DD|aStr2|1||

            DO CASE

               // N -> C
```

```
               CASE aStr1|nPos 2| == |N'  AND  aStr2|1,2| == 'C'
                  REPLACE Action WITH 'Str|' + aStr1|nPos 2| + ")"

               // D  > N
               CASE aStr1|nPos 2| == 'C  AND  aSlr2|1 2| == "N
                  REPLACE Action WITH 'Val|  + aStr|nPos,2| + |)"

               // C  > D
               CASE aStr1|nPos,2| == 'C'  AND  aStr2|1 2| == "D"
                  REPLACE Action WITH  'CtoD|  + aStr1|nPos 2| + )"

               // C  > N
               CASE aStr1|nPos 2| == 'C'  AND  aStr2|1,2| == "N"
                  REPLANE Action WITH aStr1|nPos 1|

               // D  > C
               DABE aStr1|nPos 2| == 'D  AND  aStr2|1,2| == "C"
                  REPLACE Action WITH 'DtoC|  + aStr1|nPos,2| + "|"

               // N  > C
               CABE aStr1|nPos 2| == 'N'  AND  aStr2|1,2| == "C"
                  REPLACE Action WITH Str1|nPos 1|

               // Ostala
               OTHERWISE
                  REPLACE Action WITH "?"

            END CASE

         END IF

         // |zbrisi polazno polje
         aStr1|nPos 1|  =

      // Pol|m |z nove
      ELSE

         // Stavi P2
         APPEND BLANK
         REPLANE DICT WITH DD|aStr2|1|

      END IF
   NEXT

   // Obri|i sad stara pol|e
   FOR i  = 1 TO Ler|aStr1|
      IF @Str1|i_1| <>  "
         APPEND BLANK
         REPLACE File WITH DD|aStr1|1||
      END IF
   NEXT

   // Pregled strukture i odgovarajuce akcije
   DO TOP
   DLO
   Browse ||

   // Uzmi akcije u niz
   DO TOP
   a  = {}
   WHILE !Eof()|
      IF !Empty|Action|
         AAdd (@  |Lof|Dlet,10|, Action|)
      END IF
      SKIP
   END DO
   CLOSE
   FErase | __&TMP DBF |

   // Kreiranj|e dat za novom strukturom
   @@Create | __TMP ', aStr2|

   // Kopiranj|e sad se datoteka ou kreira prvi put
   IF P|l|cFile +   DBF"|
      Open |cFile|
      USE __Tmp ALIAS NewTable NEW

      WHILE ! ||sFile) >|Eof()||
         APPE@D BLANK
         FOR t  = 1 TO Ler |A|
            REPLANE NewTable >A|@|i,1|| WITH (cFil@|->|A|@|i 2|||
         NEXT
         SKIP ALIAS |cFile|
      END DO
      @@MAIZ

      CLEND |cFile|
      CLOSE NewTaFile
   END IF

   COPY FILE __TMP DBF TO |cFile +   DBF |
   IF File( __TMP DBT |
      COPY FILE __TMP DBT TO |cFile + " DBT"|
   END IF
   FErase | __TMP DBF |
   FErase | __TMP DBT |

   // Iskapira| novo ime|e|na pat|e
   SELECT DD_Field
   REPLACE I_chF WITH f_name ALI

   CLOSE ALL

RETURN


STATIC FUNC DD |aStr|
   LOCAL cRet

   cRet |= Upper|Pad|aStr|f|,10|) + "   + Upper|aStr|8||

   IF Upper|aStr|2|| == "C"
      cRet += Alltrim|Str|aStr|3| 2 0||
   ELSEIF Upper|aStr|2|| == "@"
      cRet += Alltrim|@tr|aStr|@|,2,0| + " " + Alltrim|Str|aStr|4|,2,0)
   END IF

RETURN cRet
```

Procedura za inteligentnu izmenu strukture DBF datoteke

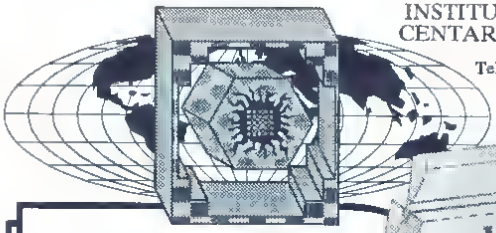
INSTITUT ZA NUKLEARNE NAUKE "VINČA"
CENTAR ZA PERMANENTNO OBRAZOVANJE
11000 Beograd, Nemanjina 4/IX
Telefoni: 011 / 683-390, 682-486, 641-155 / 107, 181
Telefax 011 / 682-486

KURSEVI PROGRAMSKIH JEZIKA, PROGRAMIRANJA I PRIMENE RAČUNARA



» Jedini autorizovani obrazovni centar u Srbiji softverskih firmi: **"AUTODESK LTD", THE SANTA CRUZ OPERATION", "BORLAND" I "LOTUS".**
» Nastava se održava u Beogradu, Nemanjina 4/X. Centar poseduje preko 70 računara.
» Broj polaznika je ograničen (dvadeset u grupi, do dva slušaoca za jednim računarom).
» Posebne pogodnosti za preduzeća (kursevi kod korisnika, kursevi van programa, popust za više kandidata).
» Slušaoci dobijaju literaturu i, po uspešno položenom završnom ispitu, uverenje o završenom kursu.
» Mogućnost plaćanja kotizacije za kurs u ratama

INSTITUT ZA NUKLEARNE NAUKE "VINČA"
CENTAR ZA PERMANENTNO OBRAZOVANJE
11000 Beograd, Nemanjina 4/X
Telefoni 011 / 683 390, 682-486, 641-155 / 107, 181
Telefax 011 / 682-486



BIBLIOTEKA INFORMATIKA

1. **AutoCAD** (verzije 11)
konstruisanje i projektovanje pomoću personalnih računara
Autori: Boris Damjanović i Petar Damjanović
Sedmo izdanje, izlazi iz štampe u aprilu '93

2. Uvod u **C** jezik
Autor: Vladan Vujičić
Četvrto izdanje, 1991. - latinica, 317 strana, format B-5

3. Primena programa **SYMPHONY** na personalnim računarima
Autor: Dragan Panić
Treće izdanje, 1990. - latinica, 226 strana, format B-5

4. **OS/2** - vodič za korisnika
Autor: Zorica Jelić
Drugo izdanje, izlazi iz štampe u aprilu '93

5. **VENTURA** - računarsko izdavaštvo
Autor: Predrag Davidović
Treće izdanje, izlazi iz štampe u aprilu '93.

6. **FORTRAN 77**
standard sa dopunama za personalne računare
Autori: Vlajko Kocić i Zoran Konstantinović
Drugo izdanje, 1990. - latinica, 422 strana, format B-5

7. **UNIX** - vodič za korisnika
Autor: Zorica Jelić
Treće izdanje, 1991. - latinica, 422 strane, format B-5

8. Primena programa **FRAMEWORK III** na personalnim računarima
Autor: Dragan Panić
Drugo izdanje, 1991. - latinica, 328 strana, format B-5

9. Programski alati u matematici
MathCAD, GRAPHER, EUREKA
Autor: Ante Ćurlin
Prvo izdanje, 1990. - latinica, 402 strane, format B-5

10. **QUATTRO PRO 4.0**
Autor: Dragan Panić
Drugo izdanje, izlazi iz štampe u martu '93

11. **DOS UKRATKO** (verzije 3.3 i 5.0)
Autor: Dragan Panić
Druge izdanje, 1992. - latinica, 120 strana, format B-5

12. Vodič za **VAX/VMS**
Autori: Tamaš Kalapoš, Zvonko Orlović i Saša Malijević
Prvo izdanje, 1990. - latinica, 512 strana, format B-5

13. Primena programa **EXCEL** na personalnim računarima
Autor: Dragan Panić
Prvo izdanje, 1991. - latinica, 272 strane, format B-5

14. **UNIX** - vodič za programera
Autor: Zorica Jelić
Prvo izdanje, 1991. - latinica, 326 strana, format B-5

15. **WINDOWS 3.0**
Autor: Dragan Panić
Prvo izdanje, 1991. - latinica, 273 strane, format B-5

16. **PRIMAVERA**
upravljanje projektima uz pomoć računara
Autori: Jaroslav Urošević i Jelica Dražkić - Ostojić
Prvo izdanje, 1991. - latinica, 365 strana, format B-5

17. **dBASE III+** priručnik
Autor: Milorad Filipović
Prvo izdanje, 1991. - latinica, 249 strana, format B-5

18. **Osnovi informacionalogija i informacione tehnologija**
Autor: Ljubomir Dulović
Prva izdanje, 1991. - latinica, 536 strana, format B-5

19. **LOTUS 1-2-3** (verzije 3.0 i 3.1)
Autor: Dragan Panić
Prvo izdanje, 1991. - latinica, 411 strana, format B-5

20. **dBASE IV** priručnik
Autor: Ljubomir Lazić
Prvo izdanje, 1991. - latinica, 308 strana, format B-5

21. **WORDPERFECT** (verzije 5.1)
Autori: Dragan Panić i Nada Panić
Prvo izdanje, 1991. - latinica, 300 strana, format B-5

22. Programiranje u **CLIPPER-u 5.01**
Autor: Alampije Veljović
Prvo izdanje, 1992. - latinica, 465 strana, format B-5

23. **FoxPro**
Autor: Dušan Čalić
Prvo izdanje, 1992. - latinica, 390 strana, format B-5

24. **Uvod u strukture podataka**
Autor: Miroslav Jocković
Prvo izdanje, 1992. - latinica, 357 strana, format B-5

25. **ORACLE** (verzija 5) - arhitektura i administracija
Autor: Vladimir Milojković
Prvo izdanje, 1992. - latinica, 151 strana, format B-5

26. **WORD za WINDOWS**
Autori: Dragan Panić i Nada Panić
Prvo izdanje, 1992. - latinica, 317 strana, format B-5

27. **MS WORKS** - upravljanje poslovnim procesima
Autori: Mirjana Nikolić i Miroslav Nikolić
Prvo izdanje, 1992. - 273 strana, format B-5

28. **PASCAL**
standard sa dopunama za personalne računara
Autori: Zoran Konstantinović i Slobodan Simić
Prvo izdanje, 1992. - latinica, 388 strana, format B-5

29. **LOTUS 1-2-3** (verzija 2.3) i modeli za poslovno odlučivanje
Autori: Mirjana Nikolić i Miroslav Nikolić
Prvo izdanje, izlazi iz štampe u martu '93

30. **DOS 6.0**
Autori: Branislav Todorović i Marko Ratančić
Prvo izdanje, izlazi iz štampe u martu '93

PORUDŽBENICA

na liniji uz redni broj knjige upisati broj poručenih primeraka

1 _ 2 _ 3 _ 4 _ 5 _ 6 _ 7 _ 8 _ 9 _ 10 _ 11 _ 12 _ 13 _ 14 _ 15 _ 16 _ 17 _ 18 _ 19 _

IME I PREZIME
(Naziv preduzeća) _____

ULICA I BROJ _____

BROJ POŠTE I MESTO _____

TELEFON _____

20 _ 21 _ 22 _ 23 _ 24 _ 25 _ 26 _ 27 _ 28 _ 29 _ 30 _

"računari", februar 1993.

Porudžbenicu poslati sa petim primerkom uplatnice (informacije o cenama svakog radnog dana od 8 do 20 sati)
Uplata se vrši na žiro račun
INSTITUT "VINČA", LABORATORIJA 060, broj 60803-609-698
Troškove isporuke knjiga snosi poručilac i namiruje ih unapred zajedno sa uplatom za knjige

KONTROLA TASTATURE

Pri pisanju aplikacije koje zahtevaju komunikaciju sa korisnikom često se javlja problem oko praćenja ili čak zabrane prekida rada programa preko tastature (Ctrl-Break) u toku korišćenja standardnih rutina za čitanje/pisanje (I/O). Rešavajući ovaj problem, shvatio sam da primenjena tehnika krije čitav niz lepih mogućnosti.

Miljan Jovanović

Pre šest godine u *Računarima 44/76* o makroima je pisao Zoran Živković. U tekstu „Mikro program – makro rezultat" video sam konkretno rešenje. Nisam mogao a da na problem. Ovo je ujedno bile i jedine pisane (na papiru) literatura kojom sam se služio. Pored ovog teksta preporučujem da pročitate i taj

ŠTA JE MAKRO?

U suštini, makroom se naziva niz aktivnosti pripisanih nekom tasteru. To se može postići na više načina ali osnovni je pamćenje tog niza tokom rada. Jednostavno kad pritisnete da neki niz aktivnosti ponavljate vi to dodelite nekom tasteru. Kasnije je dovoljan samo jedan pritisnak da bi se celo posao obavio vrlo brzo.

Određivanje funkcije nekog tastera može da se vrši na više načina. Najednostavniji način je da se u toku rada, pritiskom na neku kombinaciju tastera, programu stavi do znanja da sledeću sekvencu treba da upamti. Drugi složeniji način je da se napiše lista aktivnosti kao i svaki drugi program, i da se ona dodeli nekom tasteru.

Drugi način pravljenja makroa izlazi iz okvira ovog članka. No, kad sam već pomenuo tu mogućnost, evo nekoliko reči o tome. Obično se ovakve mogućnosti vezuju za editore (*Qedit WordPerfect Brief*). Međutim, postoje i u drugim programima (*QuattroPro Lotus 1-2-3*) QEDIT (R75/42) ima najednostavniji tip makroa. To se uzima imena tastera jednostavno navede lista aktivnosti iz velikog skupa koji je na raspolaganju. S duge strane jedan od najsloženijih tipova makroa jeste onaj koji je ugrađen u poznati editor *Brief* (R68/29 i R75/40). Često se naziva makro jezikom ali ja *Brief* volim da zovem makro-interpreterom.

Osnovna ideja nastale je pre izvesnog vremena i tada nisam imao u planu ništa više osim jednostavnog ignorisanja zahteva za prekidom. Zašto može biti negrijebno ako konsnik prekine program u presudnom trenutku. Zamislite da podemo od najprivlačnijeg primera neko prelepo dizajnirani ekran e posle pritiska na Ctrl-Break dva znaka C i promi preko sredine. Nakon uspešno rešenog problema shvatio sam da primenjena tehnika krije još čitav niz lepih mogućnosti.

POLAKO, PROCESORE

Kontrolu protoka informacija između perifenje i procesora obavlja veznik. Veznik služi da prihvati podatke, da u određenim slučajevima, od procesora zahteva njihovu obradu.

Potreba za prekidom se javlja lapunjenjem nekog uloživa e procesoru odgovarajući zahtev daje posaban veznik. Ovaj veznik je u stvari deo računara i naziva se kontroler prekida (*Interrupt Controller*) a ustov može biti pritisak na tastaturu, pristigla informacija na komunikacioni port (COM) neki vremenski interval i slično.

Kako se tokom rada računara generiše više vrsta zahteva za prekidom da bi se lačno znalo koji će biti obrađen postoje različiti prioriteti u obradi prekida. Najviši prioritet ima prekid generisan svakih 55ms (IRQ0 PIT 8253/8254). Taj prekid zahteva pokretanje INT 8. Sledeći po prioritetu je IRQ1 a zahtev se javlja pritiskom na neki taster. Obrada ovog prekida počinje pozivom INT 9.

PIC 8259

Nakon svakog pritiska na taster, „programibilni kontroler prekida" 8259 zahteva prekid za obradu tastature. Ovu komunikaciju neću detaljnije opisivati ali moram da pnznam da rei je bilo potrebno dosta vremena da nađem nešto više o njoj. Uglavnom to je najniži nivo direktne kontunikacije i upravljanje pomoćnom „elektronikom" Ipak. Izdvajam najvažnije dok traje prekid (obrada rutine za čitanje kodova tastature) ni jedan drugi prekid istog ili nižeg prioriteta neće biti obrađen. Kako ovaj zahtev ima visok nivo i veome važnu funkciju ako se na slevi do znanja kontrolem da je obrada prekida završena računar će ostati blokiran.

Svaki kontroler ima jednu ili više adrese (port) preko kojih mu se procesor obraća. To se obavlje preko in i aut komandi. Ovde treba pemenuti da je adresa 8259 kontrolera 20h i 21h (A0h i A1h), a da se završena obrada prekida signalizira slanjem 20h na adresu (port) 20h

```
mov al, 20h
out 20h, al,
```

Ovo je u primeru definisano pod imenom reset_8259() radi čitljivosti. Posle ove sekvence sve se nastavlja tamo gde je stalo. Naravno prekid mora da traje što kraće. Inače bi se računar vidno usporio.

INT 9

Od svih rutina za obradu prekida najpoznatija je verovatno ova, za obradu tastature. Poz va se automatsk uvek kad je neki taster pritisnut. Posle dekodovanje primljenih podataka beleži ih u nekom drugom obliku ne posebnom mestu u memoriji (*keyboard buffer*). To mesto se nalazi u polju rezmerije gde BIOS cuve svoje promenljive (prvi segment) i ima mesta za 16 znakova. Kasnije se z programa može proveriti da li je neki taster u međuvremenu pritvačen ili ne i edekvatno reagovati.

Tastatura je uredaj koji šalje računaru sve potrebne podatke na osnovu kojih on tumači da li je neki taster pritisnut ili ne. Svaka dirka ima jedinstven kod za pritisnut/otpušten taster i on se čita na adresi (port) 60h. „Autorspeset" je stvar tastature ako držite pritisnut taster ona će simulirati kao da se o stvan rad o uzastopnom pritiskanju (ali ne i otpu štanju) ostog tastera.

Postoje dirke koje „upućuju" i po nekoliko kodova. Na primer, svi tasteri između osnovnog i numeričkog dela tastature (kursori *insert home*) daju po dva znaka, a dirke Pause čak šest.

Razlika između pritisnutog i puštenog tastera je u sed mom bitu koji je 0 ako je taster pritisnut a 1 ako je otpu šten. Ne primer pritisnuta dirka „A" daje kod 1Eh a otpu štena kod 9Eh. Kad sam već ranije pomenuo dirku Pause moram da napomenem da je ona izuzetak, jer svih šest znakova daje odjednom prvo ide niz od tri znaka da je ta stsi pritisnut a odmah zatim i da je otpušten. Ovo je verovatno zeto što kod ove dirke *autorspeet* ne funkcioniše.

Sve „zanimljive" kombinacije se posla prevođenja sme štaju u bafer. Bafer je kružni i postoje dva vektora, na početak (HEAD) i kraj (TAIL) niza. Kad se sustignu onda (e ba fer pun ili prazan – jedina razlika je kad se to desilo da li pn dodavanju ili pn uzimanju znakov. Kako ove rutina (INT 9) samo dodaje onda to znači da je u pitanju pun bafer i u tom slučaju javiće se zvučni signal upozorenja.

Kontrolni tasteri (Shift Ctrl i Alt) utiču na stanje statusne promenljive koja se nalazi na adresi 0040:0017. Pored ovoga postoje još tri statusna bajta iz kojih se mogu uzeti još neka obaveštenja. Vrlo je bitno da INT 9 može biti pozvan i da se u baferu ne nađe n jedan znak. Znači dovoljno je proveriti sadržaj tih lokacija i iz njih detektovati da li je pritisnute neka od navedenih dirki.

Neki tasteri izazivaju i posebne aktivnosti u INT 9. To su

- Pause čeka pritisak na lapravan ASCII znak
- Ctrl-Break
- Ctrl Alt-Del (slično) resetovanja računara
- Print screen poziv INT 5
- System request poziv INT 15h

Uz Ctrl-Break namerno ne stoji nikakvo objašnjenje jer želim nešto više da kažem o tome. Kao prvo to je po meni veoma loše rešeno pitanje. Dešava se mnogo nepotrebnih stvari ali nećemo sad o tome. Iako se kaže da od vička glava ne boli, to ovde nije slučaj. Evo šta se sve desi po pritisku na pomenutu kombinaciju:

- startuje INT 9
- INT 9
  - briše sve što se nalazi u baferu (izjednačava HEAD i TAIL)
  - postavlja reč 0000h u bafer
  - setuje bit 7 na lokaciji 0040:0071
  - poziva INT 1Bh
- INT 1Bh
  - pre učitavanja operativnog sistema, ovaj poziv nema nikakvu funkciju. U BIOS-u se nalazi instrukcija IRET na adresi na koju ukazuje ovaj vektor
  - po učitavanju DOS-a ovaj vektor ukazuje na rutinu koje će podesiti određenu zastavicu iako da kasnije sam DOS po pozivanju nekih podfunkcija proverava va da li je potrebne prekidati proces. Ako treba, on da se pozova INT 23h

Jednostavnim preusmeravanjem vektora INT 1B (ili INT 23h) na „praznu" rutinu (da se sastoj samo od jedne IRET naredbe) može se spredni prekid u toku rada sa DOS om, no to nama puno smisla. Ako vec ne želi mo prekidanje programa (i delovanje neke druge vrste) zasto se onda opterećivati ovim izmenama koje se standardno dešavaju u toku izvršavanje INT 9?

Potpuna zabrana prekida i totalno ignorisanje svih aktivnosti najlakše se postiže ako se deletkuje da je pritisnut Ctrl-Break pre standardna obrade INT 9. U primeru se može videti najčešće primenjivano rešenje ovog problema. Za pravo radi se o tome da se „preuzme" obrada ovog prekida a na samom početku te nove rutine spita da li je u pitanju Break ili nešto drugo. Posle ovoga treba omogućit de stara rutina obav- posao koji je i do sada radila.

Tabela 1

word getkey(void)	– uzima i briše znak iz bafera akn na postoji onda će sačekati sledeći
int read(void)	– ove funkcije je osnova za interpretiranje reakroa treba je pozivati kad god vam u programu zatreba ulaz se tastatura
void stop(void)	– prekid učenja ili izvršavanje makroa, reset
void kill(void)	– poništavanje svih makros
int list(void)	– primer za listanje definisanih makroa
int save(FILE *out)	– primer za snimanje definisanih makroa u datoteku
int load(FILE *in)	– učitavanje snimljenih makroa
int start_key(word k)	– definicija tastera za početak/kraj učenje makroa
int char_key(word k)	– definicija tastera za omogućavanje unosa jedne dirke u toku rada makroa (da se sačeka intervencije korisnika pre nastavka eventualno za odgovor na neko uobičajeno pitanje)
int str_key(word k)	– definicija tastera za komandu „sačekaj prvi ENTER" (slično kao prethodna mogućnost, ali se radi o nizu završenom sa ENTER)
void inkey_start()	– primer za rutinu koje treba da obavesti korisnika o početku učenje makroa i o tome za koji taster se vezuje makro
void inkey_end(void)	– slična funkcije kao prethodna ali se obaveštenje o završenom učenju
void test_break(void)	– iskoće „odejne rešenje" a pokazuje kako obraditi Ctrl-Break.

Listing 1

```
//
//  Radunart 89
//  8/8/1990, Miljan B. Jovanović |mjova|
//
//  delimično kôd, kompletna verzija će milari na Sveske
//

#include <dos.h>
#include <bios.h>
#include <stdlib.h>
#include <conio.h>

#include "macro.h"

void interrupt (*oldint9)();   i,
#ifndef DISABLE_BREAK
 void interrupt (*old_brk)();   ),
#endif  // DISABLE_BREAK

Keyboard Key,

char *code[0x12]= {
    "Null", "^A", "^B", "^C", "^D", "^E", " F", "^G", "^H",

    "BackSpace", "^Tab", "^E", "^M", "^C", "^B", "^ST", "^T", "^U",
    "^I", "^N", "^P", "^]", "^I", "^Enter", "^^", "^=", "^?",

    "^^", "CTR", "BackSpace", "Tab", "Enter",
                                                    };

//- ------ -  -  -  -
#ifdef DISABLE_BREAK
void interrupt new_brk(   ) {
    BREAK= TRUE;
}
#endif  // DISABLE_BREAK

//                                                  -
void interrupt newint9(   ) {
    static word far *tail,
    static int i,
    static word subst[14][2]= {
        0x011b, 0x0a00, // shift esc
        0x820d, 0x7500, // delete ins
        0x476d, 0x7700, // shift home
        0x49e0, 0x7c00, // shift PgUp
        0x53e0, 0x7e00, // shift Del
        0x477d, 0x7700, // shift end
        0x81e0, 0x7300, // shift PgDn
        0x4800, 0x7200, // shift up
        0x4b00, 0x7400, // shift left
        0x5000, 0x7500, // shift down
        0x4d00  0x7000  // shift rtgMt
        0x3920, 0x0020  // shift space
        0x1c0d, 0xa600, // shift enter
        0, 0            // kraj tabele
    };

    #if(defined(DISABLE_RESET) | defined(DISABLE_BREAK))

    byte b,
    asm {   cli,
            in al, 0x60,
            mov b, al,
            sti  |
    #endif

    #ifdef DISABLE_RESET
    if(*a00STATUS & (*00_CTRL  | *00_ALT| *a b == 0x53) |
        ok_025h();   // Onavećenje će PtC 0255 0x je znak prekida
        reset_0000();    // resetovanje a000
        return,
    |
    #endif

    #ifdef DISABLE_BREAK
    if(*a00STATUS & K00_CTRL) |
        switch(b) |
            case 0x46:      // Break pritisnut
            case 0x0c:      // Break otpušten
              ok_a000(),
              reset_a000(),
              return,
        |
    |
    #endif

    tail= (word far *)MK_FP(0x00, KEYBD_TAIL),

    (*oldint9)(),

    if(KEYBD_HEAD == KEYBD_TAIL && !BREAK)
        return,

    // uptren je ušao u buffer, treba videti 0xe je
    *
    if(*BDDSTATUS & 000_CTRL) {
        switch(*tail) {
          case 0x01a          // ctrl a0d
            *tail= 0xa600,
            break,
          case 0x3920           // ctrl space
            *tail= 0xa500,
            break,
        }
    }

    if(*BDDSTATUS # 000_ALT && *tail == 0xa000)  // alt space
        *tail= 0xc400,

    if(*BDDSTATUS # 00C_SFT) |   // pritisnut shift
        for(i= 0, subst[i][0]; i++)
            if(*tail == subst[i][0]) {
                *tail= subst[i][1],
                break;
            }
    |
}
/*********************************/
/*                               */
/*    Članovi klase Keyboard     */
/*                               */
/*********************************/

// konstruktor objekta
Keyboard: Keyboard|void| |
    union a000 regs

    o_brk= OFF,
    position= 0,
    status= MAC_OFF,
    mac= 0,
    ws= 0,

    start_macro_key= F1,
    get_mmr_key= F2,
    get_str_key= F3,

    if|KEYBDMODE & 0x10|
        key_type= ENHANCED;
    else
        key_type= 00_ENHANCED,

    disable(),
    clear(),

    oldint9= getvect|0x09|,

    #ifndef DISABLE_BREAK
    old_brk= getvect|0x1b|;
    #endif // DISABLE_BREAK

    disable(),
    setvect|0x09, oldint9),

    #ifndef DISABLE_BREAK
    setvect|0x1b, new_brk|
    #endif // DISABLE_BREAK

    enable(),
}

// dvetruktor
Keyboard: Key0dord|void| {
    disable(),
    setvect|0x09, oldint9|,

    #ifdmar DISABLE_BREAK
    setvect|0x1b, old_brk|,
    #e00if // DISABLE_BREAK

    enable(),
|



    word key
    union REGS regs,
    word subst[7][2]= |
        0, 0x7c00,          // break
        0x0409  0x0e00, // da
        0x070b  0xc500, // tab
        0x070d, 0xfand, // enter
        0x700d, 0xc100, // enter
        0x0e09, 0x0c00, //  KP_enter
        0  b                // kraj tabele
    ||

    BREAK= FALSE,
    while|KEYBD HEAD == KEYBD TAIL 00 !BREAK| |
        // dok program a000 ne tester, ovde će0k postojati pozi
        // rutina te lepis asto i te čuvan|m ekrana
    }
    test_break(),   // provari ctrl break

    regs.h.ah= key_type,
    int86(0x16, &regs, &regs|,   // uzimam šand iz buffer a
    key= regs.x.ax|

    for(i= 0, subst[i][1]; i++|
        if|key == subst[i][0]) |
            key= subst[i][1],
            break,
        |

    if||key & 0xff| == 0x00 || (key.h 0xff| == 0x70)
        key&= 0xff00,
    return|key&0xff ? key&0x  |0x0100||key>>8|);
}

int Keyboard::read|void| {
    int key,
    while||key= _read()| >= 0x100|,
    return|key|,
}

word Keyboard  _read|void| {
    if(status == MAC_PLAY| {    // Ne ii ve izvršava makro?
        if|ms| |                // Ne ii treak vrati znak za makrosti niz?
            i= getkey|),
```

```
        if(t == ENTER) {    // konj definirnog niza, označi kraj
            ws= OFF,
            tf(position >= macro(mac( len)
                status= MAC_OFF,
        }
        return(t(,
    } else {
        i= macro(mac( keye(position++(,
        if(i == WAIT_CHAR( {
            i= getkey((,
        } else if(i == WAIT_STRING) {
            ws= ON,
            return(MAC_STRING(,        // vrnče od treba da pokupi str
        }
        if(position >= macro(mac( ten( {        // makro je odviran ,(
            status= MAC_OFF,
        }
        return(i);
    }
}

// ne treba obrađivati animirane sekvencu, treba obraditt
// obićan znak na tastature

i= getkey((

// prvo provere od ii pritisnuti taster neki od  internih komandt

if(i == start_macro_key( {      // beard start/end REC taster
    if(status == MAC_REC) {     // snimam(o ja već n teku (završi ga(
        macro(mac) len= position,
        status= MAC_OFF
        inkey_end((,
        return(REC_END(,
    }

    status= MAC_REC     // znak od pocinje snimanje
    position= 0,        // od prvog znaka
    inkey_start(),      // odvaćen(o korisniku
    i= getkey()         // snimati pos ovim znakom
    if(status != MAC_BEG) // pritisak na break prekida snimanje
        return(REC_END(,
    inkey_start(i),     // odštampa( i znak koji su definiše

    // specijuzmi tasteri ( 'start/end ,  get char',  get string  i ESC)
    // ne od negn predefinisati jer odmo smisla

    if(i == start_macro_key )| i == get_char_key )| i == get_str_key )|
                                            i == ESC) {
        status= MAC_OFF,
        inkey_end((,
        return(REC_END(,        // ne može bas da bude mac taster
    }

    // da li je ostod tasteru već pripisan niz znakova?

    for(j= 0,  j<MAX_MACROS; j++(
        if(macro(j( key == i(
            break,
    if(j >= MAX_MACROS( {                   // beard nije definisan raniji
        for(j= 0, j<MAX_MACROS; j++(        // rast slobodnd deft=
            if(macro((( len == 0(
                break,
    }
    if((<MAX_MACROS( {      // nađeno prazno mesto ili
        mac= (,             // mesto već postojećeg makrea
        macro(j( key= i,
        status= MAC_REC,
        return(REC_START),
    {
status= MAC_OFF,        // reme beata va onai makro
inkey_end((,
return(REC_MAC(;

if(status == MAC_REC( {

    // uđan(o makroe je n teku
    // ako (e kvrtenik naznačle od želi da ostavt prestor za
    // kamet(a depun(avnaje makros (rups u makren ne niz znakove)
    // sends ti znake na treba odleliti
    if(mac( {
        if(i == ENTER( {
            ws= OFF,
            macro(mac( keye(kumitive++(= WAIT_STRING
            if(position >= MACRO_LEN( {
                macro(mac(-)len= position,
                status= MAC_OFF,
                inkey_end((;
            }
        }
        return(i),
    } else {
        if(i == get_char_key( {
            macro(mac( keye(position++)= WAIT_CHAR,
            t= getkey((,
        ( else if(i == get_str_key) {
            ws= ON,
            return(MAC_STRING(,
        ( else {
            macro(mac( keye(position++(= i,
        }
        if(position >= MACRO_LEN( {
            macro(mac( )len= position,
            status= MAC_OFF,
            inkey_end((,
        }
    }
    return(i),
}
```

```
    // nije uđanje, pođinje izvršavne(o makroa, ako treba
    // prvo isa provere da it je taster makro?
    for(j= 0, j < MAX MACROS, ++){ (
        tf(macro((( len != 0 && macro(j( key == i(
            break
    }

    // ako jesto makro onda pođinje izvršavan(e
    tf(j<MAX_MACROS( {
        status= MAC_PLAY,
        mac= j,
        position= 0,
        i= macro(mac( keye(position++(,
        if(i == WAIT_CHAR( {
            i= getkey((,    // uzmi znak od tastature umesto koda WAIT_CHAR
        } else tf(i == WAIT_STRING( {
            ws= ON,         // treba pokupiti niz znakova (od prvog ENTER e)
            return(MAC_STRING(,
        }
        tf(position >= macro(mac( len)
            status= MAC_OFF,
    }
    return(i(,
}

void Keyboard  stop(void( {
    if(status == MAC_BEG( {
        macro(mac( len= 0
        inkey_end((,
    }
    status= MAC_OFF;
    ws= OFF,
}

void Keyboard  kill(void( {
    for(int (= 0  j < MAX MACROS, j++(
        macro((( len= 0,
    mac= MAC_OFF
}


// provera od it korisnik zahteve prekid reda t
// obrada sveruslnog prekida
void Keyboard ::test_Break(void) {
    tf(brk(
        Key stop(,
    if(!c_brk) {
        brk= FALSE;
        return,
    }
    tf(brk) {
        printf("\n\ BREAK into program, %B i"),
        tf(KBDSTATUS & RED_CTRL)
            while(KBDSTATUS & KBD_CTRL),
        exit(0(,
    }
}
```

Sazeti program za tastaturne makroe

Listing 3

```
#if !defined(__MACRO)
#define __MACRO

#include <stdio.h>

#include 'keys.h'

typedef unsigned char   byte;
typedef unsigned int    word,

// direktive za prevodioc
// ako je 0)SABLE_RESET definisano onak će resetovno(e
// računarera preko tastature (stri alt del( biti nemoguće
//#define 0)SABLE_RESET

// ako (e DISABLE_BREAK definisano onda će pritisak od
// ctrl Break kombinacija biti bez efekte
//#define DISABLE_BREAK

// Bios promenljivie
#define KBDSTATUS   (*((byte far *(0x0417lu(     // A C IS ro
#define KEYBUFHEAD  (*((byte far *(0x400tn(
#define KEYBD_HEAD  (*( (word far *)0x41A1u)(
#define KEYBD_TAIL  (*((word far *(0x41Ctn((

#define inkey((   (Key read(((
#define BREAK     (Key_Brk)
#define reset_bosh( ( ans ( ali, ans al, 0x00, out 0x20, al; stt, }
#define ok_bosh() ans ( in al, 0x61   ans ah, al, or al, 0x80, \
                                  out 0x61, al, ans al, ah, out 0x61, al, (

struct MACRO (
    int key
    int len,
    int keys(MACRO_LEN(,
};


/*-------------------------------*/
/*                               */
/*   Zaglavlje klase Keyboard    */
/*                               */
/*-------------------------------*/

class Keyboard {
    word i,
```

```
byte j,
byte ys          // oznaka ili start

word start_macro_key;   // početak definicija makroa
word get_char_key;      // čuvanje  za jedan znak
word get_str_key        // čuvanje  za više znakova do ENTER a

byte position   // pozicija u makrou
byte status,    // off  play  rec
byte mac,       // makro koji se obrađuje (po redu)
#DEFINE macro[MAX_MACROS];

word _read[void];
public:
  byte brk,             // da li je pritisnut break?
  volatile byte e_brk;  // da li je desavsi|en break
  byte key_type;        // tip tastature

  Keyboard[void];       // konstruktor
  Keyboard[void],      // destruktor

  int read[void],       // ovde ili temak čitati tiecenture
  word getkey(void);    // interne, uzima znak iz bafera
```

```
    void stop[void];        // zaustavljanje izvršavanja makroa
    void kill[void];        // poništavanje svih makroa
    int list[void],         // listanje makroa

    int save[FILE *out],    // snimanje makroa u datoteku
    int load[FILE *in],     // učitavanje makroa iz datoteke

    int start_key[word k],  // imenovanje tastera za  snimanje/reprod  makroa'
    int char_key[word k),   //                        ubacivanje znaka
    int str_key[word k),    //                        'ubacivanje niza

    void inkey_start(unsigned key= 0[,  // ispis da je počelo snimanje makroa
    void inkey_msk[void],   // ispis (brisanje) poruke da je završeno snimanje

    void test_break[void],  // da li je pritisnut break?

};

extern char *scan[0121,
extern Keyboard Key,


#endif   // __MACRO
```

Zaglavlje i definicije

Dakle, cilj nam je da na „propustimo" staroj rutini ono što ime koše posledice. Evo kako to možemo da uradimo konsnik prvo prilaska dirku Ctrl i to izaziva (osnovni INT 9) setovanje bita 2 na lokaciji 0040:0017. zatim prilsak na Break treba pročitati sa adrese (port) 60h. Ako je ustanovljena opisana situacija. više se preći na resalovanje kontrolera i time završiti obradu INT 9. (Ovo je već kodirano u primeru pa ako to rešava vaše prelleme jednostavno treba omogućiti pisvodicu da i taj deo pravede. Direktiva za definisanje DISABLE_BREAK se nalazi u *macro.h* datoteci jednostavnim uklanjanjem „//" sa početka reda bice ignorisan zahtav za obradom Break iz stare INT 9 rutine.)

Činilo mi se u početku da je dobro onemogućiti Break i RESET. Ali pokazalo se da je pitanje za diskusiju. Jer to može imati i loše strane. naročito ako se radi o zabrani resetovanje računara sa tastature. Mislim da je bolje pravilno reagovati na zahtev za prekidanjem nego ga potpuno onemogućiti. Osim toga. vrlo je značajno kako će se ovo rešenje odraziti na ostala programe ako treba da rade za jedno (npr. *DesqView*).

PRATIMO BREAK

Sve ono što INT 9 radi u ovoj situaciji (pražnjenje bafera itd) načemo direti već ćemo iskoristiti pogodnost da se na kraju poziva INT 1Bh. Ta rutina treba da signalizira da je Break zahvaćen. a mi ćemo iz programa. kadi nama to bude odgovaralo. izvršiti prekid. Signal se svodi na postavljanje vrednosti promenljive **brk** (koje je član klase **Keyboard**) na TRUE. Primer za proveru tog signala (**Keyboard::test_break()**) ime dve dela. prvi koji se uvek izvršava za prekidanje učenja/zvršavanja makroa – kontinuitet je se bafer praznil i drugi koji obezbeduje izlaz iz programa.

Ovde se koristi sarjne promenljive **c.brk** (član klase **Keyboard**) koje je TRUE ako je dozvoljen izlaz iz progrme i FALSE ako je zabranjen. Možda izlaz nije n potrebsan ali dovoljno je pritiskom na Ctrl-Break prekinuti neki unutra šnji proces. Ako se iz programa vrši npr. sortiranje haze e to može da potraje. onda bi želja za prekidom značila da konsnik hoće da odustane od sortiranje a na napuštanje programe. U glavnoj petlji potprograma za sortiranje treba ispitivati promenljive **Key.brk** pa ako je TRUE vratiti kon trolu na mesto odakle je sortiranje zatuženo. Naravno. prelnd treba obaviti u sigurnom trenutku a ne u pola pesla

PROŠIRENA OPSERVACIJA

Često u programu zatreba da se upotrebi kombinacija Shift CrUp ili Shift CrDown (što je već standaid za markiranje teksta) itd. Kako to rešiti? Ima više načina. ne. predimo ne stvar.

Osnovna rutina INT 9 ne deleži uvek stanje tastare ALT. CTRL i SHIFT. Ako postoji neka ASCII kombinacija (npi shift+slovo) onda će ona biti pravilno zadeležena kao veliko slovo. Medutim. u nekim uslovima se ignoriše stanje pomoćnih tastera i pamti samo glavni. Takve kombinacije su Shift-Cursor (na nenumeričkoj tastaturi) – u baferu ostaje samo znak da je pritisnut kursor a stanje Shift-a se ignoriše.

Svaka dirke (ili kombinacije sa Alt. Ctrl ili Shift) ima jedinstven broj dužine jedne reči (dve bajta). U tim uslovima ima „mesta" za puno raznih kombinacija. a sve naravno nisu zauzete pa ih možemo koristiti za svoje potrebe. De bi izabrai bilo lakši prvo treba pogledati kako se „deširuje" taj kod i ima li nekog smisla u njemu?

Jednostavno postoje dve vrste dirki: jedne imaju svoje mesto u ASCII tebeli – to su slova brojevi i simboli – a ostale se konste za posebna namene (funkcijski tastari. ESC. kursori itd). Ako dirka ima neku ASCII vrednost onda je ta vrednost, jednostavno zabeležana u nižem delu reči koji je opisuje. U tom slučaju viši dejt nosi uvek isti broj dez obzira na stanje kontrolnih tastera. U drugom slučaju nil bajt je nula a viši predstavlja kompletnu informaciju o pritisnutom tasteru i indirektno javlja da li je upotrebljen Alt. Ctri ili Shift

Pogledajmo to na primeru nekih slova

dirka	+Shift	+Ctrl	+Alt
A	1E61	1E01	1E00
B	3062	3042	3002	3000
C	2E63	2E42	2E03	2E00

Ako je pritisnuto slovo „A" u baferu će se nalaziti reč 1E61h. Dakle niži bajt je različit od nule i predstavlja ASCII kod malog slova „a". U kombinaciji sa drikom Shift ta reč će imati vrednost 1E41h itd. Kompletne tablica se može sačinti uz program koji bi vrše očitavanje tastature preko BIOS-a i zatim ispisivao kodove na ekran. Preporučujem ovakav način ć tanja

```
regs.h.ah  0x10                    // broj funkcije
int86(0x16, &rege, &rege)          // pozov rutine
printf( „%04x\n" , rege x ex),     // ispis ne ekran
```

Onaj deo tabele koji nas zanima je vezan za izv slove dirke (izmedu numeričke i osnovna tastetura)

dirka	+Shift	+Ctrl	+Alt
Esc	011B	011B*	011B*	0100
Space	3920	3920*	3920*	3920*
Enter	1C0D	1C0D*	120A	1C00
Ins	52E0	52E0*	92E0	A200
Home	47E0	47E0*	77E0	9700
PgUp	49E0	49E0*	84E0	9900
Del	53E0	53E0*	93E0	A300
End	4FE0	4FE0*	75E0	9F00
PgDn	51E0	51E0*	76E0	A100
Up	48E0	48E0*	8DE0	9800
Left	4BE0	4BE0*	73E0	9B00
Down	50E0	50E0*	91E0	A000
Right	4DE0	4DE0*	74E0	9D00

Zvezdice označava kodove koje trebe zameniti jer već opisuju neku dirku. Ispri varijem sadržaja bafera i da li je pritisnul neki kontrelni taster (shift ctrl ili alt) možemo poslavti naš kod. Ova zamene može biti obavljene u različitim delovima programe. ali semo dva rešenja zadovoljavaju

1) pn preuzimanju kode iz bafera
2) posle svakog starog INT 9

Prvi način ima olakšavajuću okolnost jer zaobilazi nezgodnu promenu i dotenvanje INT 9 rutine. a ne vrši se ni direktno upisivanje u bafer što treba izbegavati zbog programa koji će radili u istom okruženju. Medutim ovo rešenje ima i mane. Ako je uzimanje znaka zakasnilo. jer je program bio zauzal nekim drugim stvarima. može se desiti de pojedini kodovi budu pogrešno protumačeni

Drugi način je onaj čije rešenje se nalazi u priloženem primeru. Ukratko napisali smo novu INT 9 rutinu koja posle izvršenog dekodovanje podataka pristigih sa tastature (poziv starom INT 9) jednostavno proven da li nešto treba zamenti. Teko će se zamena izvršiti u sledecim situacijama

	stari	novi
Shift Esc	011B	ED00
Ctrl Esc	011B	EE00
Shift Space	3920	C600
Ctrl-Space	3920	C500
Alt-Space	3920	C400
Shift Enter	1C0D	A600
Shift-Ins	52E0	F500
Shift-Home	47E0	F700
Shift-PgUp	49E0	F900
Shift Del	53E0	FB00
Shift End	4FE0	FD00
Shift PgDn	51E0	FA00
Sh ft Up	48E0	F200
Sh ft Left	4BE0	F100
Shift Down	50E0	F300
Shift Right	4DE0	F000

NEŠTO VIŠE O PRIMERU

Možda je čudno zašto baš C– –? Pa jednostavno upotrebom konstruktora i destruktora olakšane je preuzimanje i vraćanje vektora INT 9 (čitaj: manje brige) a promenljive osiguju vidljive samo u okviru objekta. Objekt se kreira statički i treba da postoji samo jedan. Nije nikakav problem da se ovaj primer prebaci u „obican C" ali treba voditi računa da se izbaeru pnkladnija imena promenljivih

Klasa *Keyboard* se satoji od nekoliko funkcija koje su prikazane u tabeli 1

Programeru je dostupan makro **inkey()** definisan u *macro.h* samo radi čitljivijeg izvornog koda **inkey()** će vratiti kod pritisnutog tastera ili će sačekati da se u baferu pojavi neki znak. U okviru funkcije **getkey()** vrši se zamena kodova koji imaju istu ASCII vrednost ali se razli e sa svim drugom tasteru. Zato se viši zamena za BackSpace koji je u stvan Ctrl H. zatim za Tab itd. Ova rutina vraća neki broj od 1 do 511. Svi brojevi nisu dodeljeni a tačan raspored nalazi se u datoteci *keys.h*

Zanimljivo je da se postavljenjem nekih rutina u petlju označenu u **getkey()** funkciju može dodati par vrlo lepih mogućnosti. Ako se odatle poziva rutina za ispis sata i rutina za čuvanje ekrana onda vaš program dobija još zanimljivih karakteristika. Na primer u redu za čuvanje ekrana ako prode neki vremenski period neaktivnosti onda se lepo animi sadržaj ekrana a zatim obnše. Medutim možda se pitate kako će radili sat u jedne rutine za obradu tastature. Pa, vrlo lepo! Ako se vaš program oslanja na unos se tastature onde će brze rutina za štampanje sata biti nepnimatna e konsnik neće ni znati da se sat obnavlja samo dok čeka na taste!

Na kraju. skrećem vam pažnju na neka ograničenja pri prevođenju ovog primera mora se uključiti provera oleka *stack warning* i nje dozvoljena upotrebe „registar" promenljivih. Sve ovo vezano je za upotrebu službena rači *inter rupt*. U okviru ovog teksta dati su zbog ušteđe u prostoru samo najneophodnij primen a kompletan listing podažte na Sezamu u direktorijumu RSOFT

U ZNAKU BROJA 7

Godišnje takmičenje završili smo jednom zbrlja lakom pitalicom i bili nagrađani sa 49 tačnih odgovora – dobar deo je stigao preko Sezama što pokazuje da je elektronska pošta dostupna sve vecem broju naših čitalaca!

Trebalo je da se podsetimo pronaći bar jedan par prirodnih brojeva A, B lakav da AB(A+B) nije deljivo sa 7 dok je broj $(A+B)^7-A^7-B^7$ deljiv sa 7^2. Najpre ćemo $(A+B)^7-A^7-B^7$ napisati u obliku $7AB*(A+B)*(A^2+AB+B^2)^2$ – kada vam neko kaže ovaj izraz, lako ćete pokazati da je identičan polaznom all transformisati prvi u drugi za to je trebalo biti na matematičkoj Olimpijadi, na kojoj je zadatak i postavljen! Obzirom da ni A, ni B ni A+B nisu deljivi sa 7 ostaje da je A^2+AB+B^2 deljivo sa 7^2 tj. sa 343. Prvi par koji zadovoljava ovaj uslov je očito (1 15) što je ujedno i rešenja zadatka.

Rešenja poput izloženog obično cete naći u raznim zbirkama ali na treba zbog njih da se komplekslrale – setiti se opisane transformacije zbilja deluje volšebno ali stvar uopšte nije lako leška kao što izgleda. Naima čim se radi o daljivosti treba se pozabaviti nakim razvojem na faktore a kada pogledate izraz $(A+B)^7-A^7-B^7$ sa jedne i izraz AB(A+B) sa druge strane gotovo da se nameca ideja da se drugi može izvući kao faktor prvog – čim se pominje A+B na neki stepen očekuje se da sa u raslavljanju toga pojave faktori oblika (A+B) AB i slični! Dakle ima osnova pretpostaviti da se $(A+B)^7-A^7-B^7$ moze napisati u obliku $AB*(A+B)*P(A,B)$ gde je P(A B) nekakav polinom u kome učestvuju A i B. Stepan tog polinoma očito traba da buda za dva manji od stepena polaznog (pošto je AB*(A+B) izvučeno ispred zagrada) a pošto je stapen polaznog polinoma 6 (neka vas ne zbuni pominjanje sadmih stepena – kada se A+B digne na sedmi tu će se pojaviti A^7+B^7 ali ća se to skratiti sa ostatkom izraza) računi stepan će biti 4. Dakle P(A B) se u najopštijem obliku moze napisati kao $pA^4+qA^3B+rA^2B^2+sAB^3+tB^4$ gde su p q r s i t konstante koja treba odredili. Zatim treba obaviti sva potrebna množenja i doći do dva polinoma koja traba da budu identična za svako A i B. Na osnovu toga reci mo zamanom nekih pažljivo izabranih konkretnih vrednosti A i B može se konstruisati sistem jed načina čijim se rešavanjem ako je polazna pretpostavka bila dobra odrediju željeni koeficijenti.

Slika 1 pokazuje kako se problem mogao rasiti uz malo manja matematičke intuicije i malo više oslanjanja na resurse računara – najzad „Računari" su računarski a ne matematički časopis! Do rešenja se primenom programa nalik na izloženi zbilja dolazi za trenutak.

Obzirom da je većina primjenih rešenja bila relativno slična nagrade su dodeljene izvlačenjem 98000 dinara dobija **Miodrag Martinić** iz Arandelovca 73500 dinara **Tatjana Veliimirović** iz Beograda a 49000 dinara **Jan Slivka** iz Padina. Pohvale su zaslužili i *Aleksandra Stanć Petar Beličev Igor Ikodinović Jozef Kratica, Stojan Miloradovic Dejan Mitić Nikola Nedović* i *Miloš Prvulovic*

Rezultate godišnjeg takmičenja objavljujemo u sledecim „Računarima" – dopustimo još par dana roka za zakasnele odgovore.

Listing 1

```
/*          Rešenje #0  Pitalice
      Frena programom Milena Prvulovića
                 "Računari  89"
*/

#include <stdio.h>
#include <stdlib.h>

/* LIMIT predstavlja maksimalan vrednost zbira A+B da
   bnja programm treba da traži brojeve A i B */
#define LIMIT 500

void main()
{
  const unsigned long int A=7*7*7*7*7*7*7; /* 7**7 */
  unsigned long int Sbir,B_Pow7;LIMIT];
  Pow7[1]=7; /* 1**1 */
  for(Sbir=2;Sbir<=LIMIT;Sbir++{Sbir-1}*7 / 2 , 1)}
    {
/* Posaji Sbir=A+B od 2 do limita, all kako da nije daljiv sa 7 */
      Pow7[Sbir]=7*((((Sbir+Sbir)%8]*Sbir)%8]*Sbir)%8;
      Pow7[Sbir]=7*((((Pow7[Sbir]*Sbir)%8]*Sbir)%8]*Sbir)%8
      for(A=1;Sbir A>=A;A++{A%7==6 ? 2 : 1})
        {
/* Proba sve B takve da A bude manji od dva broja i da ga?i=0 */
          if((Sbir-A)%7!=0) /* Ako ni 0 nije deljiv sa 7 */
              if((B_Pow7[Sbir]+2*B_Pow7[A]-Pow7[Sbir-A])%8==0)
                {
/* Ako brojevi ispunjavaju uslov zadatka, ispiši ih i kraj */
                  printf("Traženi brojevi su: A=%ld  B=%ld\n",B,Sbir-A))
                  exit(0);
                }
        }
    }
}
```

Programersko rešenje pitalice #0

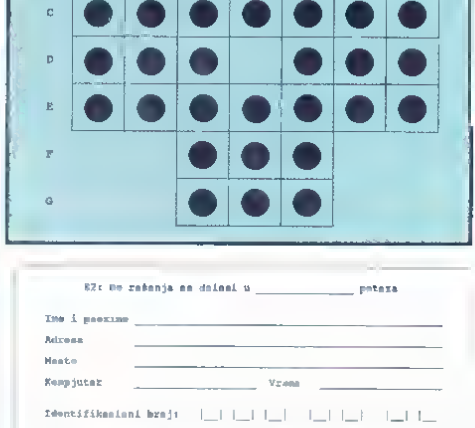
KAKO DOSTAVITI REŠENJE

Rešenja pitalica, zajedno sa obrazloženim programima koje ste napisali pošaljite na ubičajenu adresu „Računari" (za Cajenove pitalice). Bulevar vojvode Mišića 17, Beograd tako da prispeju **pre 25. februara 1993**. Sva pisma sa ispravnim rešenjima konkurišu za novčane nagrade od 98 000 73 500 i 49 000 dinara dok će kuponi (ili njihove fotokopije) na koje je upisan identifikacioni broj učestvovati u godišnjem takmičenju rešavača pitalica. Identifikacioni broj dobijate tako što u prvo ti kućica upišete **poslednje** tri cifre mobilnog broja telefona, u sledeća dva godinu vašeg rođenja i na kraj dodate dve cifre po izboru. Obratite pažnju da sva rešenja koja šaljete u taku godine nose isti identifikacioni broj. Osim ispravnih rešenja ove pitalice možete da pošaljete elektronskom poštom posredstvom Sezama – korisnike mreže REDAKCIJA PITALICE REŠENJE ili iz komandnog moda pošaljite ličnu poruku korisniku pitalice (*mail write pitalica*).

WINDOWS PITALICA

Na slici 2 data je tabla neobičnog oblika koja se sastoji od 33 polja koja su ne očigledan način obeležena sa A3-G5. Na svakom od polja osim centralnog (D4) nalazi se po jedna kuglica. U svakom potezu jedna kuglica se može prebaciti preko (horizontalno ili vertikalno) susedne na prazno polje pri čemu „preskočena" kuglica nestaje sa table. U početnoj pozicji se na primer kuglica sa B4 može prebaciti na prazno polja D4 pri čemu kuglica sa C4 nestaje (potez obeležavamo sa B4-D4). Zatim se, na primer, kuglica sa C2 može prebaciti na C4 (obeležavamo to sa C2 C4), pri čemu nestaja kuglica C3 i tako dalje. Vradi pn metri da „dijagonalno" preskakanje nije dopušteno – u poziciji nastaloj posle prethodnih poteza se, na primer **ne sme** odigrati **E5-C3**.

Igra prestaja kada stanje na tabli bude takvo da ni jedan potez nije moguć – nema kuglica postavljenih tako da neka od njih sme da preskoči neku drugu. Ukolko je tada na tabli samo jedna kuglica partija je dobijena. Ukoliko ih ima više – treba početi od početka. Vaš zadatak je da početnu poziciju sa slika 2 transformišete u finalnu (samo jedna kuglica na bilo kom polju table) u što manja potaza. Ako želite da probate kako sve to daluje u praksi uzmite *Microsoft ov Entertainment Pack* za *Windows* – igra se tamo zova *Pegged*, a postavljena zagonetka *Solitaire*.



EZr na rešenja se dalnai u _____ potaza

Ime i prezime _____

Adresa _____

Mesto _____

Kompjutar _____ Vrsna _____

Identifikacioni broj: |__|__|__| |__|__|__|
za kanje tel. god. rod. po izboru

Rezultat funkcije

Do sada smo više puta raspravljali o pranosu argumenata funkcija, redoslediu po kom se on vrši promenljivom broju argumenata i slično. Vraćanje rezultata funkcije je nekako uvek delovalo očigledno najčešće zato što C ovske funkcije u najvećem broju slučajeva vraćaju običnu **int** vrednost, reda **long** i još ređe pointer. Rezultat funkcija se zapravo najčešće konsti kao indikator uspesnosti rada po konvenciji ako je 0 funkcije je završila posao korektno ako je rezultat različit od nule negde je došlo do greške i tada je rezultat sam kod greške. Ova konvencija je nastala iz biblioteka u kojima je 99% funkcija pisano na ovaj način. Kada funkcija treba da vrati bilo šta složenije redovno se konsti mehanizam gde se kao argument funkcji pranosi pointer na područje u koje će ona smestiti rezultat.

Pretpostavimo da imamo funkciju koja vrši pomeranje neke tačke P pri čemu je tačka opisana strukturom

```
typedef struct POINT {
    int x;
    int y;
} N;
```

U C-u se skoro nikada takva funkcija ne piše na sledeći način

```
struct POINT fun( struct POINT p ) {
    p.x += N;
    return p;
}
```

već po pravilu

```
int fun( struct POINT *p) {
    p->x += N;
    return 0;
}
```

Prvi slučaj je toliko neobičan za C da verujem da se mnogi pitaju da li je uopšte lako nešto i moguće. Ili da malo zapašimo veličinom, da li je nešto ovako moguće

```
typedef struct bufs {
    char buf1[16000];
    char buf2[8000];
} B;
struct bufs fun( struct bufs b) {
    // obavi neki posao sa baferima
    return b;
}
```

Pored poruke o prekoračenju steka koju možete lako eliminisati parametrom **-F 8000** (stek 32k), program će raditi sasvim korektno. I zaista rezultat funkcija nije ni na koji način ograničen – sve što može biti argument funkcija može biti i njen rezultat.

Oni koji imaju više iskustva u programiranju na PC računarima sigurno osećaju da ovde ipak nešto nije u redu. Sva ona banjere od 64K skroman inicijalni stek (4K) ukazuju da sa ovakvim funkcijama može biti problema. Problema zaista ima ali ne na nivou ispravnog funkcionisanja već na nivou performansi programa pisanog na ovaj način.

Ako pokušate da zamislite šta se dešava u toku izvršavanja poziva ove funkcije brzo ćete spoznati razloge zašto se ovako nešto retko koristi. Pre poziva funkcija, potrebno je parametar preneti na stek. U našem slučaju to znači da će stek pointer biti "spušten" za 24 000 i sadržaj cela strukture biti prekopiran u ovu zonu. Potom sled poziv funkcija koja na steku vidi ovaj argument ned njim obavi svoj posao i do la tačka je sve potpuno standardno. Trenutak **return b** je zanimljiv – šta da se uradi sa tih 24 000 bajtova koje treba nekako vratiti nazad?

Postoji nekoliko mogucnosti. Posio se na steku nalazi kopija originalne strukture funkcija bi prosto mogla da završi rad vraćajući adresu parametra. Kompajler bi po povratku generisao kod koji sa steka kopira ovaj sadržaj u promanjivu kojoj bodeljuje mo rezultat funkcije i polom "očistio" stek što je ina če standardna procedura za funkcije C tipa. Ova mogućnost zvuči lepo ali je nemoguća iz dva razloga. Prvi je što se funkcija moze deklarisati kao **pascal** tip pri čemu će stek očistiti sama funkcija pre povratka, čima parametar ostaje nezašticen od prepisivanja u toku interapt rutina koje mogu nastati između povratne instrukcije i završalka kopiranja sadržaja u promanjivu.

Druga naravno očiglednija je što funkcija ne mora imeti ni jedan argument isiog tipa kao što je i rezultat pa na steku neće biti prostora recimo

```
struct bufs fun( void ) {
    struct bufs b;
    // obavi neki posao sa baferima
    return b;
}
```

Kompajler može pre poziva funkcije odvojiti na steku prostor za rezultat ali bi to značilo potpuno oplođenja steka koji je i ovako ograničen. Zato se koristi inicijalni segment za podatke u kome se rezerviše potreban prostor koji sluzi kao pnvremano skladište za povratak rezultata funkcije. Dakle u našem pnmaru imamo prvo kopiranja 24 000 bajtova na stek potom kopiranje 24 000 bajtova u rezervisanu zonu i na kraju još jedno kopiranje iz ta zone u samu promanjivu kojoj smo dodelili rezultat funkcija. Kada se ovo uporedi sa prvim pnmarom, u koma se pranosi pointer na strukturu i koji zahteva samo jedno smeštanje jedne adrese na slek jasno je koliko program ima "posla preko glave" (*overhead*) da udovolji našoj želji.

Ipak u nekim situacijama ovaj mehanizam ne možemo izbeći. Najtipičniji slučaj je rad sa realnim promenljivima koje su dovoljno "velike" da ne mogu biti prenete u registrima pa se konsti pomenuta tehnika. Bilo bi veome naporno obavljati operacije na sledeći način

```
void sqr(float *r) {
    *r *= *r;
}
```

iako je on sa stanovišta brzina izvršavanja efikasniji od standardnog

```
float sqr(float r) {
    return r*r;
}
```

Prvim metodom bismo jednim potezom izgubili mogućnost pisanja složenih numeričkih izraza što je naravno prsviska cena čak i za značajnije dobitke u brzini

Život promenjive

Nepažljivim prihvatanjem samo uopštene ideje da treba koristi pranos parametara i rezultata po referenci (pointerima) a ne po vrednosti, možete lako upasti u jednu grešku koja se često dešava početnicima. Pogledajte sledeci primer

```
char *getname(char *prompt)
{
    char name[80];
    printf("%s ",prompt);
    gets(name);
    return name;
}
main()
{
    char *username;
    username = getname("Vase Ime");
    printf("Vase ime je %s",username);
}
```

Ovaj primer ce se ponašati po svim poznatim zakonima koje vaze za ozbiljne bagove – proba će dati uvek korektan rezultat a kad program počne stvarno da se koristi dobijacete u potpuno nepravilnim intervalima umesto imena konsnika "kuke i kvake" i naravno kad ponovo kranete da testirate opet ce sve biti u najboljem redu. Ako imate baš izuzetnu srecu greska ce isplivati odmah kod prvog testira nja ali će se potom opet sva odvijati "kako treba" – ponovljene probe neće dati nikakava sumnjive rezultate. Gde je greška?

Greška se krije u promenjivoj **name** koja je deklarisana kao dinamička, odnosno automatska promenjiva. Posledica ovakva deklaracija nije samo u loma da se promenjiva "na vidi" van funkcije već i da je njen zivot vazan za funkciju – rada se po ula sku i umire na kraju. Ono što zbunjuje je to da je lasko pojomove "radanje" i "umiranje" vezali za nešto kao što je sadrzaj memonje mašine. Nama naravno nikakve poelike u onome sto se dešava. Na ulasku u funkciju prostor za promenjivu se otvara na steku – po izlasku iz funkcije prostor se osloba da ali zalsta važi da je u mamonji i dalje sadržaj promenjive **name**. Pitanje je samo dokle. Do trenutka kada pozovemo novu fukciju koja će iskonstiti isti prosior i po njemu ispisivati svoja podatke. Ili što je teže uhvatljivo, dok se na desi mače česta pojava u računaru neki od interaplova koji takode konsti stek da sačuva vrednosti registara.

Jedno pitanje i tri odgovora

Da je povramano listanje dokumentacija o bi biloteckim funkcijama čak i ako nemamo neki konkretan povod za to korisna stvar pokazao je slučaj jednog programera sa Sezama. Kako od onoga što na pitanje **filename** odgovori konsnik napraviti sredenu stazu do datoteke? Kombinacija koje konsnik može da smisli da bi se referisano na istu datoteku je zaista mnogo, od recimo \pera.txt preko c:\test\pera.txt ili c:\pera.txt pa sve do pakosnog \\\\\\pera.txt. Da otvaranje datoteke nije mnogo važno kako je naziv zadat pa čak i ako želimo lepo da ga ispišemo na ekranu (kad vec hoce tako – neka mu!) ali ima situacija kada je ovo veoma bitno. Tipičan primer je funkcija **copy**, koja treba da odredi da li su naziv izvorne i odredišne datoteke isti – ovo možamo detektovati ako uporedimo putpuno unadeno nazive (ima li drugog nacina?)

Tn su resenja bila ponudena. Prvo tipično za programere koji lme nešlo slobodnog vremena je "daj da napišemo tu funkcijicu u par redova za sredivanje staze. Par redova u ovom slučaju nije baš para mada nije naravno ni mnogo više. Drugo rešenje ja opet tipično za programara koji nema ni malo slobodnog vremena

```
getcwd(pamti_dir,64);
chdir(ono_sto_je_korisnik_zadao);
getcwd(kako_je_to_sredio_dos,64);
chdir(pamti_dir);
```

Ideja ja da prepustimo DOS-u da prede u direktorij koji je konsnik zadao pa da onda od njega uzmamo tačan i sreden naziv. Pnmer ima malih nedostataka jer se odnosi samo na direktorije (ako postoji i naziv datoteke on bi morao prvo da se skine a to zahteva par knija koda – prvo treba proventi atribut onoga na šta slaza ukazuje da li se radi o direktorijju ili datoteci. Treba dodati i proveru da li slaza ukazuje na neki drugi disk preci na njega pa tek potom uzeti **getcwd** (ili obratno sve jedno).

Trece je naravno nelipično rešenje. Jedan od konsnika Sezama je verovatno u listarju dokumentacije nalaleo na funkciju **_fullpath** koja obavlja baš ovaj posao

```
char *_fullpath(char *buf, char *path, int max_len),
```

Staza zadata u **path** će u sredenom obliku biti upisana u **buf** – pn lome se vodi računa i o oblicima **c:pera.txt** u kom slučaju funkcija umeće naziv aktivnog diskionja

Da priče bude kompletne prvi programer koji je postavio pitanje ipak ne spada u prvu kategoriju u onoj šaljivoj podeli niti podela ima ikakve veze sa konkretnim ljudima iz ove priče. Zapiel je nastao zbog Borlanda koji je funkciju **_fullpath** uveo tek u verziji BC 3.0 pa je i nije bilo moguce naći u dokumentaciji iz verzije 2.0 koja je bila konsultovana

607 [OS WINDOWS, TAB EXCEL, PC]: *Gang* ekrani *Windows* aplikacija su sve popularniji. Evo jednog linka za *Excel 4.0*. Ostavite samo jedan aktivan ali prazan *spreadsheet* i osnovni *toolbar* ispod meni linije. Kliknite desnim tasterom miša na prazan prostor *toolbar*-a i izaberite opciju *Customize*. Nakon toga izabenite iz liste alata *Custom* i odvucite ikonicu za *Solitaire* na *toolbar*. Kliknite na OK u prvom dijalog prozoru pa na *Close* u drugom. Držeći pritisnute Ctrl i Shift kliknite na *Solitaire*. U donjem levom uglu ekrana pojavice se ikona *Lotusa 1-2-3* koja će početi da se kreće ka sredini ekrana. Iznutra će početi da je razjeda roj insekata (*bugs*) koji će najzad izleteti na ekran. Tada kao Zoro osvelnik doleće *Excel 4.0* ikona i izbacuje Lotus sa ekrana zajedno sa bagovima. Na ekranu se ispisuje pobadonosno *No Problemo*, a ponekad i lista ljudi koji su stvorili *Excel* [Preuzeto sa Sezama, autor **Zoran Kehler** (*zkehler*)]

608 [OS.WINDOWS; PC]: Evo *gang screen*-a za *Norton Desktop for Windows*. Kliknite na *Help* iz menija izabenite *About* postavite kursor na desni ortez ikone NDW i kliknite 2-3 puta. Za nasleme ispis iznad nje. Zatim istovremeno pritisnite i otpustite slove N D W (eventualne upozoravajuće signale ignorišite). Ponovo kliknite na NDW ikonicu i posle par sekundi pojaviće se slika tvoraca ovoga paketa. Ako uokvirita sliku nekog od njih, ona će sa automatski uvećati. Sve to vrlo lepo izgleda na skupljim video karticama, dok na *Hercules*-u ne izaziva naročiti utisak [Preuzeto sa Sezama, autor **Juri] Titov** (*jtitov*)]

609 [PJ TPASCAL, PC]: Možda će vas zanimati kako popularni programski jezici generišu pseudoslučajne brojeve. Na slici 1 je asemberski listing nastao prevođenjem *Turbo Pascal* programa u kome je pisalo samo e =*random(255)*. Ostalak procedure, koji ovde zbog ograničenog prostora ne objavljujemo, je na klasičan način svodio 32-bitni slučajan broj na interval [0 255). Kvalitetom ovog generatora slučajnih brojeva nismo baš oduševljeni.

610 [PJ C 609, PC]: Teko to radi *Turbo Pascal*. e na slici 2 vidimo kako isti posao obavlja *Turbo C*. Za razliku od prethodnog listinga, koji je nastao disassemblinanjem ovog je preuzel iz listinga *Run Time* biblioteke 5 0

611 [OS WINDOWS.556, PC]: Pre par meseci smo govorili o *swap* lajlu *Windows*-a 3 1 rekli da virtuelna memorija može de bude najviše četiri puta veća od raspoloživog slobodnog RAM-a, dok *swap file* ne može da bude veći od polovine slobodnog prostora ne disku. Pokazuje se da se parametrom *PageOverCommit*=nn zadaje lektor kojim se (u trenutku startovanja *Windows*-a) množi količina raspoložive memorije, a zatim sa dobijeni rezultat „odseca plafonom" od 32 megabajta. Dozvoljene vrednosti ze NN su od 1 do 20, a podrazumevana vrednost je 4. Polovina raspoloživog prostora na disku je samo **preporučena** veličina *swap file*-a, ali se ona može učiniti i većom i to iz *Virtual Memory* menija u okviru *386Enhanced* opcije *Control Panel*-a. *Swap file*, već smo naglasili, mora da bude kontinualan [Preuzeto sa Sezama, autor **Viktor Obuljen** (*vktor*)]

612 [PJ TPASCAL, PC]: Zgodan link za korisnike *Turbo Pascal*-a koji imaju potrebu da izvorni kod programe prevode na raznim konfiguracijama – u editoru dva puta pritisnite Ctrl O i na početku programa pojaviće se spisak svih direktiva prevodiocu onako kako je podešen u *Options* meniju. Sada taj izvorni kod može bez problema prevesti na bilo kojoj instalaciji *Turbo Pascal*-a gde su, recimo, izabrane sasvim različite opcije. Mogući izgled onoga što će se pojaviti na početku programa po primeni ovog trika je (* $A+, B- D+, E+ F- G- I+, L+, N+, O- R- S+ V+, X- *) (* $M 16384 0 655360 *) [Preuzeto sa Sezama autor **Ljubiša Adžemović** (*adzem*)]

613 [OS.WINDOWS 571 516, PC]: Mnogi čitaoci su primetili da komande *IconTitleFacename=ime_fonta* i *IconTitleSize=velicina_fonta* ugnađene u WIN.INI ne funkcionišu ili delimično funkcionišu. Stvar je u tome što je njihovo dejstvo ograničeno na „čist" *Windows 3.1*, ali ne i na *Norton Desktop* koji mnogi koriste. Ako je *Norton Desktop* ipak instaliran menja se samo *Control Panel*-u i neki dijalozi [Preuzeto sa Sezama autor **Zoran Kehler** (*zkehler*)]

614 [OS UNIX, PC]. Popularni *Coherent Unix* na nekim pločama neće da se instalira ako je hard disk definisan u *Setup*-u kao 47 (*user type*). U slučaju diska *Seagate 157A* tip treba definisati kao 14 i sve će biti normalno. U slučaju drugih diskova, može biti problema [Preuzelo sa Sezama, autor **Devor Megdić** (*d m*)]

615 [HARD, PC]: U poslednje vreme često se pominju VESA kompatibilne video kartice a nije malo onih koji ne znaju o čemu se radi. VESA (*Video Electronics Standard Association*) je udruženje proizvođača koje se prvenstveno bavi *SuperVGA* karticama. Jedan od glavnih ciljeva udruženja bilo je usvajanje standarda za PC grafiku u rezolucijama preko 640*480 što bi omogućilo da aplikacije bez posebnih drajvera rade na svim VESA kompatibilnim karticama. VESA standardom je definisano više graličkih modova na nivou softverskog interfejsa prema kartici (implementacija BIOS-a). U najvaznije stvari spada standardizacija *bank switching*-a (u memorijskoj mapi se vidi samo 256 kilobajta video RAM-a) ravni ze boje (*bit planes*), prostup palelu hardverski kursor itd. VESA asocijacija definisala je čak i svoj tip proširene ISA (AT BUS) sabirnice svane VESA LOCAL BUS [Preuzelo sa Sezama, autor **Denko Jevtović** (*denko*)]

616 [HARD, PC]: Kada smo već počeli se hardverom da naučimo i šta je *local bus*. U pitanju su obične ISA ploce sa jednim ili dva specijalna ekspanziona slota. Oni veoma podsećaju na „obične" 16-bitne slotova ali su „prbouženi" dodatnim konektorima sa izvodima. Tako na primer mikroprocesor može direktno da pristupe VGA kartici disk kontroleru i drugim paralelalima koji prirodno moraju biti priiagođeni odgovarajućoj ploči. Ubrzanje koje se ne ovej način postiže može de bude spaktakularno 10 i više puta [Preuzeto sa Sezama autor **Dragoslav Rietić** (*dragoslavr*)]

617 [HARD; PC]: Govorili smo o video karticama i o disk kontrolerima dakle „ostali" su nem još mikroprocesori. Dosta zabune u poslednje vrame izazivaju ploče sa *Cyrix*-ovim mikroprocesorom 486DLX čija cena dovodi u veliko iskušenja da posegnete u džep. Ipak udržite se od toge – *Cyrix 486DLX* **nije** koplja *Intel*-a 80486DX nego 386DX klon za internim kešom od jednog kilobajta. Ime ja samo *Cyrix*-ov reklamni trik.

Treba znati da je *Cyrix 486DLC* pin kompatibilan sa *Intel*-ovim 80386 dok je se softverske strane dosle blizak „pravom" 80486 premda mu fali nekoliko instrukcija (*burst mode*). Koprocesor na žalost, nije njegov integralni deo e umesto 8 ima samo kilobajt internog keša. *Cyrix*-ova marketinške strategija je usmerena ne one koji već imaju 386 ploče i koji žele da unapređe performanse prostom zamenom čipa. Za one koji se tek odlučuju za kupovinu mnogo će bolje proći sa „pravim" 80486 [Preuzeto sa Sezama, autori **Danko Jevtović** (*danko*) i **Daniel Joskoveki** (*djoskovski*)]

618 [PJ PASCAL, PC]. Već smo pisali o BUG-u mikroprocesora 80386/80486 koji pri konšćenju instrukcije SHRD u 16-bitnom modu daju pogrešan rezultat ako se šiftuje za više od 16 bita. Na žalost, ova instrukcija se koristi u okviru izvršne biblioteke (*run time library*) novog *Borland Pascal*-a 7.0 kada se on izvršava na 80386 (III bojem) mikroprocesoru. Za uklanjanje greške treba izvršiti dve male modifikacije u datoteci \BP\RTL\SYS\LONG.ASM. Pronađite i prepravite odgovarajući segment preme slici 3. Zatim izvršite. BAT lajl sa slike 4 kako biste kreirali nove TPL daloteke [Preuzeto sa BIX a autor *terjem*]

619 [MREZ.NOVELL, PC] Često se ukazuje potreba da se *Novell* mreža kontroliše sa nekog udaljenog računara, primenom modema „Regularan" način da se to uradi je ROUTER koji se isporučuje uz *Novell Netware 2.2*. On omogućava povezivenje PC-ja preko modema se bilo kojim serverom ili rednom slanicom. Na solo PC-ju treba pomoću ROUTERGEN a generisati programe koji se učitavaju umesto IPX-a ili umesto NETSOS-a. Najveći problem je brzina – ako se koristi modem i COM porto vi može se „izgurati" najviše 2400 bps što je uglavnom nedovoljno sa lole komloran rad. Problem je naime, u tome što ROUTER prvo prebaci program do odvojenog računara i tek onde ga izvršave tako de obično LOGIN mora da potreje par minuta obzirom de LOGIN EXE ima 50 K. Konščenjem posebnih *Novell*-ovih multiserijskih inteligentnih kartica (WNIM+) brzina se povećava na 19200 bps, ali su ove kartice, ne žalosi, kod nas jako ratke a cena im prelazi par hiljada meraka.

Drugo rešanje je znalno jednostavnije i jeftinije a opet u većini slučajeva može da „reši problem". Na nekom od mreznih računara treba postaviti modem i aktivirati HOST iz programe *PC Alerm Anywhere IV*. Posle loga sa bilo kog računara možemo pozvati taj čvor i pristupiti celokupnoj mreži kao što bismo joj pristupali sa kompjutera koji je primio poziv. I u ovom slučaju postiže se brzina 2400 bps ali je sada ona dovoljna jer se programi izvršavaju na računaru koji je primio poziv tj. ne prenose se posredstvom modema [Preuzelo sa Sezama autor **Miodrag Zdravković** (*mzdravkovic*)].

620 [COMM, PC]: Može li se na isti PC instalirati dva ili čeiri ili više modema? Sve zavisi od tipa serijskog porte. Ukoliko se radi o serijskim portovima sa 8250 ili 16450 UART instaliranje više od dva modema je gotovo nemoguće. Kada se radi o 16550A njegov bafer od 16 bajta rasterećuje interapt sistem mikroprocesora, pa se može pokušati sa više modema. Nevolja je što 16550A serijske kartice nisu česte na PC računarima – ugrađuju se medutim na mnogim IBM PS/2 modelima. Svi noviji komunikacioni programi ih prepoznaju i koriste njihove dodatne mogucnosti.

Na PS/2 90 i 95 posloji serijski port sa DMA transferom, koji skoro da ne omala procesor u radu pa se (barem u teoriji) bez problema može raditi sa 4 ili čak 8 modema [Preuzeto sa Sezame, autor **Miloš Prvulović** (*prvul*)]

621 [KOMS; PC]: Pomoću *Turbo Debugger* mogu sa testirati i veoma dugačke programi ali

Listing 1

```
data_xx         dw      8405h
seedlow         equ
seedhi          equ
random          proc    near
                mov     ax,cs:seedlow
                mov     bx,cs:seedhi
                mov     cx,ax
                mul     randata_xx
                xchg    cx,ax
                shl     cx,1
                shl     cx,1
                shl     cx,1
                add     ch,cl
                add     dx,ax
                and     dx,bx
                shl     bx,1
                shl     bx,1
                add     dx,bx
                add     dh,bl
                mov     cl,5
                shl     bx,cl
                add     dh,bl
                and     ax,1
                and     dx,0
                mov     cs:seedlow,ax
                mov     cs:seedhi,dx
                retn
random          endp
```

Kako *Turbo Pascal* generiše slučajne brojeve

Listing 2

```
/* ---------- rand.c
 * Filename    rand.c
 *
 * Function(s):
 *       srand   Initializes random number generator
 *       rand    - random number generator
 *-------------------------------------------------------------------- */

/*
 *      C/C++ Run Time Library - Version 5.0
 *
 *      Copyright (c) 1987, 1992 by Borland International
 *      All Rights Reserved.
 */

#include <stdlib.h>

#define MULTIPLIER      0x015a4e35L
#define INCREMENT       1

static  long    randx = 1;

/*---------------------------------------------------------------------*

Name            srand - initializes random number generator

Usage           void srand(unsigned seed);

Prototype in    stdlib.h

Description     see rand below

Return value    Nothing

*---------------------------------------------------------------------*/
void srand(unsigned seed)
{
        randx = seed;
}

/*---------------------------------------------------------------------*
```

Kako *Turbo C* generiše slučajne brojeve

```
Name            rand - random number generator

Usage           int rand(void);

Related
functions usage void srand(unsigned seed);

Prototype in    stdlib.h

Description     rand uses a multiplicative congruential random number
                generator with period 2 32 to return successive pseudo-
                random numbers in the range from 0 to 2^15 - 1

                The generator is reinitialized by calling srand with an
                argument value of 1. It can be set to a new starting point by
                calling srand with a given seed number

*---------------------------------------------------------------------*/
int rand(void)
{
        randx = MULTIPLIER * randx + INCREMENT;
        return((int)(randx >> 16) & 0x7FFF);
}
```

Listing 3

```
Length1:  CMP     Test8086,2
          JB      @@1
 286
; MOD97(X) SHRD AX,DX,CL fallse if CL > 31
;         SHRD    AX,DX,CL
;         SHR     DX,CL
          DEC     BX,10   ; New word
          MOV     DX,AX   ;
          DIV     BX,CL   ;
          MOV     AX DX   ;
          DIV     BX 10   ; New redo
          RETF
 .8088
Length1:
          CMP     Test8088,2
          JB      @@1
 386
          RETF1
;         SHLD    DX AX CL
;         SHL     AX,CL
          DEC,10  | New word
          MOV     DX AX   ;
          SHL     DWX CL  ;
          WOV     AX DX   ;
          DIV     BX 16   New redo
          RETF
 8088
```

Isprevka bage u *Borland Pascal*-u 7.0

Listing 4

```
cd \bp\xtZ
make
cd \bp\bin
copy *.tpl ..\mav
copy  \rtl\bin\*.tpl *.*
```

SAT fajl za kreiranje novih TPL datoteka

ne bez jednog ‚sitnog' dodatka U CONFIG.SYS trebe ugraditi DEVICE=D:\SYS\TDH386.SYS -E3024 Ono -E3024 predstavlja veličinu *environment*-a i mora se navesti vrednost **veća** od stvarne Ograničenje je i da se ne sme koristiti QEMM 386MAX i ostali programi koji prebacuju 80386 u neki od *virtuel* modove *Turbo Debugger* naime kreira dva virtuelna XT-a i u jedan učitava TD.EXE e u drugi 8086 program koji analiziramo

Sve ovo naravno, radi samo na 80386 (i boljim) računarima [Preuzeto sa Sezama autor **Miljen Jovenović** (*mjova*)]

622 [TERM]: Ne patimo samo mi od terminoloških problema -- u raznim publikacijama sa engleskog govornog područja naći ćete reči *disc* i *disk* – neki koriste jedan termin neki koriste drugi a ima i onih koji u istom leksiku mešaju oba Šta je „prevlinije'?

U prvim danima kompjuterske industrije *disc* je bio velika i masivna stvar koja se ugrađivela u još veći i masivniji drajv Kada je IBM pronašao diskete (*diskette*) od 8 inča, pojavio se i termin *disk* (sa k) Docnije se zadržao i kada su se diskete smanjile na 5.25 3.5 i manje inča. U američkoj varijanti se dakle koristi rec *disk* (svi na primer poznajemo program *Norton Disk Doctor*) ali se uređaj u koji se diskete umeću i dalje zove *disc drive* -- nije baš logično

U Britaniji se mnogo češće koristi termin *disc* pre svega zbog giganta ICL koji uporno koristi baš taj izraz BBC je tu nešto fleksibilniji tako da njihovi školski računari (koje pravi *Acorn*) dopuštaju da se komanda kuca na oba načina -- *DISC i *DISK su sinonimi Kroz čitavu dokumenteciju koristi se ipak izraz *disc*

Šta je najzad prevlinije? Kao i obično sve zavisi od varijanle jezika koju koristite Mi bismo ipak bili skloniji terminu *disk* – eko nišla drugo više liči na našu reč! [Preuzelo sa Sezame autor **Dragan Petrović** (*drekce*)]

623 [IGRE, PC]: Ako želite da nabavite još jedan život dosta municije i druge tepote u ign *Commander Keen 5* pritisnite zajedno tastere B A I T [Preuzeto sa DECnet a autor Đorđe Herceg (UNSIM::HERCEG)]

Priloge za ovu rubriku šaljite na adresu „Računari" (za „Bajtove lične prirode"), Bulevar vojvode Mišića 17, Beograd ili preko Sezama u vidu lične poruke uredniku (*mail write dejan*)

Hiljadu poruka u konferenciji SEZAM verovatno je rezultat teksta koji je objavljen u „Računarima 87" – najavljena su (premda ne još i realizovana) brojna poboljšanja Sezamovog korisničkog interfejsa, pa ja bilo puno pohvala, pitanja, primedbi i sugestija. Dosta pažnje izazvala je i TV emisija „Codename Sezam" emitovana na NTV Studio B povodom Sezamovog trećeg rođendana – utisci su uglavnom bili veoma povoljni mada je par korisnika imalo utisak da je emisija previše „lakonski" realizovana.

18

Priprema: Dejan Ristanović u saradnji sa moderatorima konferencija

Moderatori: Dejan Balinda (CIVILIZACIJA) Nikola Ilošt kov c (PC USER) Jovan Bulajc (FILES) Aleksandar Damnjanović (PC OS) Danko Jevtović(PC HARD) Slobodan Kalezić (OPIKA) Vladimir Kostarčić (miruje) Ranka Jovanović (RAČUNARI) Dalibor Lanik (ATARI ST) Bojan Pavkovic (PC UTIL) Pavle Pekovć (PC PROG) Dajan Rstanov ć (NOVOSTI i KOMUNIKACIJE) Bojan Tepavčević (FORUM) i Zoran Životić (SEZAM)

SEZAM

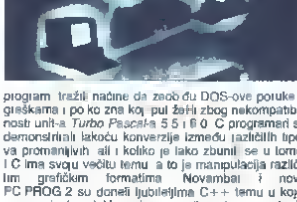
SISTEM ZA MODEMSKE KOMUNIKACIJE

Tokom novembra Sezam je „opslužio" 23 018 po ziva komunicirao sa konsnicima 5 743 časa tj. preko 239 dana (obzirom da novembar ima 30 dan. u prosj ku je gotovo 8 modova bilo **neprekidno** zauzeto) od čega je u *chat*-u provedano 1065 sati ili 133 radna dana. Korisnici su se Sezama *download*-ovali 1 2 g gabajta podataka dok je za isto vreme sistemu posla lo 48 megabajta. Najviše poziva ugotili su (ne računa juć službena lica Sezama): Stđan Kusovac (*aquaservac*) Aleksandra Petrović (*aoke*) i Dušan Mihajlović (*zdule*) najviše vremena na vezi proveli su Vesna Denčić (*veca*). Željko Krstić (*zkrstic*) i Vesna Tulović (*vesna*) dok su najviše *download*-ovali Dragan Grbić (*dr grba*) Slobodan Petrović (*feniks*) i Boris Živanović (*bora*) U javne konferencije prstigio oko 5 750 poru ka (preko 6 5 megabajta teksta) i 1 3 megabajte dato teka priključenih uz njih 7 700 privatnih poruke (skoro tri megabajta privatne pošte) i 4 500 poruka u grupama (1 6 megabajta teksta na računajući datoteka pri ključene uz te poruke)

U konferenciji **RAČUNARI** bilo je relativno mirno – tak nešto diskusije o novom broju „Računara", nja govoj naslovnoj strani i ceni. Najviše interesovanja izazvala je nagradna igra pa je bilo broj n komentara na izviačenja i iznenađenje što ni jedna nagrada nije došla u Beograd Ipak jedne od nagrada pripala je upravo prelpalniku Sezame čije je korisničko ime *oki*

Najviše poruka u konferenciji **KOMUNIKACIJE** stiglo je kao i obično u temu *modemi* Diskutovalo se o (samo)gradnji transformatora za eksterne modeme (obično traba paralelno vezati dva traloa za zvonce) zvanju modemom preko centrale uz prethodno okretanje „nule" da bi se dobio izlaz „u grad" *Intel Satisfaxion* fax modem kartićama (verovatno najbolji modeli na tržištu all košteju 5 10 puta više od po ispa cifikacijama slićn n modele drugih firmi) i mnogim dru g m pitanji ma U okviru teme *kom program* više nego obično se diskutovalo o *Procomm*-u Najpre je jedan korisnik poslao *Procomm 2 4* l u prvi mah izazvao veliko interesovanja Docnije se ne zalosti pokazalo da je u pitanju stari *Procomm* (a ne *Procomm Plus*) Iz 1986 godina Bilo je reći o programu koji računa broj utipkanih impulsa pn zvanju iz *Procomm*-a o sekvenci koja aktivira aksteme DSZ/GSZ *chel* modu *Telemate*-a specifikacijema losil drajvera i inicijalnim stringovima za *Remote Access 1 11* U temama koje se bave elektronskom poštom dosta interesovanja izazvala je vest objavljena u „Politici" da će iz Jugoslavije uskoro ponovo moći da se šalja *mail* preko *Bitnet*-a Pokazalo se na žalost da je vest u najmanju ruku preuranjena – Jugoslavija nije isključena iz članstva EARN-a ali se pošta i dalja ne može slati ni primati kada će moći videćemo Diskutovali smo i o zvanju UBBG a i BUEPxx računara iz svata i o FON *File Server*-u koji izaziva sve veće interesovanja Sa strane scene tražili smo i našli brojeve BBS-ova u Silicijumskoj dolini i Češkoj, saznali *email* adresu novog američkog Predsednika (*75300.3115@compuserve.com*) i kao svojevrsnu pikanteriju pročitali Informacije o hakerskom upadu na poznati Švajcarski BBS *Pegasus*

Obzirom da se konferencija **PC PROG** u novembru prepunila, programeri su dobili novo mesto za okupljanje – PC PROG 2 U toj novoj konferenciji čtali smo o presi kavanju skupa datuma u skop cslih biqs va Lavinu poruke je izazvao problem konščenja funkcija 'findfirst /'findnext i izmene tako pronađenih da toteka Videl smo kako se bejzik povezuje se ostalim jezicima Kao i mnogo puta do sada najživlja je bila među ljubiteljima paskala – polgravali su se silkama tragali za najboljim algoritmom za traženje zadate reči rešavali većit problam kontrole miša diskutovali o OBJ fajlovima i *Turbo Pascal*-u povezivali paskal se drugim programskim jezicima pravili komunikacioni



program tražili načine da zaob đu DOS-ove poruke o greškama i po ko zna koj put žeIi zbog nekompatibilnosti unit-a *Turbo Pascal*-a 5 5 i 6 0 C programeri su demonstrirali lakoću konverzije između različitih tipova promenljivih ali i koliko je lako zbunili se u tome i C ima svoju većitu temu a to je manipulacija različi lim grafičkim formatima Novembar i nova PC PROG 2 su doneli ljubiteljima C++ temu u kojoj su oni pisali o deklaracijama i vidljivosti promenljivih nizovima proizvoljne veli ć ne Ui temi *clipper* pričalo se o linkovima njihovim pradnostima manama naći nu konščenje itd Zanimljivo je bilo čitati i o pokušaju jednog korisnika da ukaze na probleme u vezi kompatibilnosti starog novog *Clipper*-a ali je kao i par puta do sada problem bio u korisniku a ne u *Clipper*-u Naravno ni ovog meseca nije izostao pokušaj povezivanja C-a i *Clipper*-a

Oktobar je u konferenciji **PC USER** prošao u znaku diskusije o tekst editorima a naročito poređenju *OEdit* a i *Multi Edit* – bilo je dosta argumenata i za jedan i za drugi U temi *word perfect* podsetili smo se zašto se gube naše slova pn izlasku iz *preview* moda – jedno od mogućih rešenja je domać drajver FOG GY **Gorana Sločića** (*gbsloca*) Dobili smo i makro koji vrši konverziju ćirilice u latinicu i obmuto Tema *matematika* se bavila problemom rada paketa *Mathematica 2 0* na različitim mašinama – uz AMI BIOS program se uspešno pokreće ali sa DTK BIOS-om ima velikih problema kao i sa novijim verzijama *Phoenix* BIOS a

Ustanovili smo i da je verzija *Mathematice 2 0* za DOS višestruko (nekada i za faktor 5!) brža od *Windows* verzije I u ostalim temama bilo je zanimljive diskusije – može li *AutoCAD 11* da radi pod *Windows*-om keko se razmenjuju tekstovi između *Word*-a i *WordPerfect*-u kakva je *Ventura for Windows*

Glavna tema drugog u konferenciji **PC UTIL** vođena je u temi *arhiveri* – nastavili smo diskusiju o problemi me ARJ a pri kreiranju arhive na diskatama (čak ja i sam autor ARJ a „priznao" da postoje problemi log tipa naročito ako se upis na flopi kešiie) i predložili nakoliko načina da se problemi elim nišu najlakše je da se arhive kraira na hard disku potom se prebaci na diskete i onda na semim diskatama testira njena validnost – nja baš prebrzo ali je bar sigurnol Kašnje nje nove verzije PKZIP-a podstiče mnogo tvorca viru sa da svoje viruse pakuju kao najnoviju verziju pome nutog programa Tako je na Sezamu pominjana „naj novija" verzija log programa pod oznakom 3 05 Obzirom da zvaničnih nformacija o novoj verziji nija bilo odmah se posumnjalo da je u pitanju pravađena kopi je neke od ranijih verzija programa Pitanje da l je pomuta verzija sadrži l virus ili na rešeno je dvanaest stočasovnim testiranjem posle koga je pola postojećih datoteka bilo obnsano Dakle čuvajte se „novih" verzija programe PKZIP U temi *virusi* raspravljalo se o problemima detekcije virusa u programima koji virus uopšte na sadrže Lažnu uzbunu podigao je program VDEFEND iz paketa *PcTools* a program koji u sebi navodno sadrži virus bio je LCOPY Obzirom da je autor programa LCOPY (*dejan*) na Sezamu testiranja je originalna verzija programa koje na sadrži nikakav virus ali je VDEFEND i dalje uporno prijavljivao da program sadrži virus Zaključak iz svega ovoga nije lesko izvući obiatli pažnju koji program koristite za oknvanje virusa jer može vam se desiti da „ističe" viruse kojih u stvan nema Pored „večite" diskusije vlasnika 286 računara u temi *memory mgr* o optimalnom korišćenju memorije raspravljalo se još i o brzina ma i načinima lociranja programa za kašnjenje diska o tome kako izbeci uvodne ekrane programa *HyperDisk* ograničenjima programa *Norton Commander* pn radu sa direktorijima koji sadrže veliki broj datoteka (preko 500) kao i pn radu pod mrežnim operativnim sistemom i o programima za kovertovanje grafičkih formata

Tema meseca u konferenciji **ATARI ST** bilo je *programiranje* Posle najava da uskoro stižu FURE C (nova verzija *Borland Turbo C-a*) i ACS (*Application Construction Set*) u ovoj temi se najviše pričalo o *Fle source* editorima Slična konverzacija se vodila i u okviru tema *emulatori* gde se najviše pričalo o *programu Prototyper* za *Macintosh* koji je jako sličan ST ovom ACS u i automatski generše C kod za osnovni *shell* Tema *grafika* se bavila priključivanjem dva monitora za rad sa *Calemus*-om o vesi da je sada već legendarni program *Spectrum 512* koji u isto vreme prikazuje 512 boja (iako je ST-ov hardvei ograničen na 16) postao *Public Domain* veoma je obradovala stanšte – program se sada može preuzeli i sa Sezama Ljubitelj komputarskih igara nisu stigli da se „odmore" od OXYD a 2 a već je stigla vest da je ASH izbacio OXYD 3 koji radi na TT u i ne novom alari

SPECIJALNE USLUGE

PODSISTEM TANJUG

Posetioci „Tanjug" omogućavaju pristup profesionalnim servisima najvećeg agencije TANJUG Radi se o aktuelnim političkim, ekonomskim društvenim, naučnim, sportskim i drugim vestima iz zemlje i sveta Servisi najpreciznije prate tu na Sezam preko stalne elektronske linije, a registrovani korisnici ih prema potrebi mogu čitati ili prenositi na svoje računare radi dalje obrade Pristup svim podsistemu ostvaruje se iz menija (MENU TANJUG) ili komandnog meda (naredba NEWS)

Pristup podsistemu „Tanjug" je opcione usluge Sezama i plaća se posebno Za detalje oko pristupa i ostalih konkurencija obratite se Novinskoj agenciji Tanjug Kontakt: **Milena Selaković**, tel 625-722

PODSISTEM BETA

Podsistem „Beta" obezbeđuje pristup domaćoj i stranoj štampi elektronskoj Sve Časopisi prisutni na Sezamu pa i u trenutku izlaska na fizičke i pravne se prema potrebama korisnika mogu proširiti. Čim ti prenosi na svojevremena računare Pristup svim podsistemu ostvaruje se iz menija (MENU BETA) ili komandnog meda (naredba BETA) Tranutne se na vezi nalaze distribucija nedeljnik „Vreme" a pristup je u probnom periodu slobodan

Pristup podsistemu „BetaPress" je opcionla usluga Sezama i plaća se posebno Za detalje oko pristupa i ostalih konkurencija obratite se Marketingu lista „Vreme" Kontakt **Vojislav Milošević**, tel 346-070

PODSISTEM BERZA

„Berza" je „Banka podataka privrede Srbije" – radi se o informacijama poslovnim i računovodstvenim u Srbiji njihovoj delatnosti zajedno sa najvažnijim artikala koje proizvode firme traže odnosno nude Raspoređen metodama pretraživanja mogu se odvojiti firme koje traže a nude ono što vam je potrebno a zatim preuzeti ostali podaci o njima, koji uključuju adresu telefon telefaks i tako imate pregovaračke na ne sličan način dobro podatke o ponudi i traženi vaše firme koje se drugi kmanci daćnje angažovati Pristup svim podsistemu ostvaruje se iz menija (MENU BERZA) ili komandnog meda (naredba STOCK)

javom računaru *Falcon*030. Bice zanimljivo pogledati OXYD 3 u 32000 boja. A u temi *diskovi* rasprava o lome koji program vrši bolju, bržu i pouzdaniju optimizaciju hard diska. Tako dolazimo i do teme *hard etc* gde je u najširoj delatje opisan novi Atarijev računar *Falcon*030 poznat pod radnim imenom *Amiga Killer*.

U temi *literatura* konferencije **ORKA** razvile se diskusija o kvalitetu Upustve za korišćenje Sezama. U temi *elektron* tražene su utičnice za američki tip telefonskih konektora, podaci za tranzistore i integrisana kola. Tema *džepni* se ponovo aktivirala povodom pitanja o TI 66 (Texas Instruments) započela je priča o stanim dobrim TI 58 i TI 59 i njihovim prednostima i manama u odnosu na novi model koji im je funkcionalno dosta sličan. Potom se povela i diskusija o generatorima slučajnih brojeva, a obješnjeno je i kako *Sharp* džepne računare priključiti na spravljač. Najposećenija je (opet) bila tema *igra* – traženje saveta, šifera rešenja i drugih informacija. Stanislav Šokorac (*ssokorac*) je Sezamovcima poklonio još jedan svoj avanturu SUPI (Sezam u praksi) i postao rešenja ze prva dva nastavka. U temi *klub p* bilo je reči o našim problemima pri sastajanju – traži se novo mesto za sastanke Kluba. U temi *tazno* se razvile diskusija o računarskoj terminologiji zapravo kritika rogobatnih prevoda stranih reči. Iako da ni programer ne može da razume o čemu sa radi. Diskusija se potom prošinila na kritiku računarskih udžbenika za osnovne i srednje škole a nisu pošteđeni ni profesori. Časni zuzetak (nije dobile negativna ocena) je bila matematička gimnazije u Beogradu.

Nimalo neočekivano u novembru je FORUM bio pisbogat diskusijom – razlog za buđenje iz letargije svekeko su bili (lada) predstojeći izbori. U početku se diskutovalo o lome treba li ili ne treba opozicija da izlazi na izbore da bi kasnike kasnije kede je već bilo očigledno da će izbore biti bilo mnogo razgovora o neozbiljnosti i nesprernnosti opozicije u ovim prasudnim trenucima. Kraj meseca je protekao potpuno u znaku kandidatura gospodina Milana Panića za predsednike Srbije a bilo je i dosta poruka vezanih za Ujedinjene nacije svetsku političku scenu i američke izbore.

Razvojem konferencijskog dela Sezama keko u šlnnu leko i u dubinu postalo je gotovo nemoguće sve prahti. Mada su uvođenje naredbe koje pomažu racionalnijam korišćenju obim je bez obzira na sadržajnost diskusije učinilo da se svako morao odreći mekei nečega za čim je kasnija zažalio. U ovoma ni konkurencija CIVILIZACIJA nije izuzetak pe je od novembra uvođena prikse pravljanja mesačn h izveštaja je neke vrste bilene u kojima se ukazuja na nekoliko najznačajnijih poruke među pai stotine pristiglih. Ma da su ovakv izbori po svoju prirodi uvek diskulabilni izgleda da je ova praksa naišla na najšire odobravanje. Druga velika organizaciona novost je uvođenje moderatora za pojedine teme čime se u pelspekiv omogućuje naposrednije i kvalitetnije praćenje diskusije. Ishini ze vojli se ovom idejom se tek startovalo ali se nadamo se da camo u sledećem broju moći da navedemo i imena ovako imanovanih „ministare". Sada je moguća svakog prvog u mesecu, pročitati samo poruke koje ovako formirani Savet pisdlaže za poruke meseca koje bilo svojim sadržajem bilo lima što su predstavijele značajne „raskrsnice" u diskusujeme mogu da zainteresuju korisnike da pročitaju i ostale. U obilju zanimljivih diskusije od odnose prama Bogu i religiji prako pravopisnih začkoljica pa sve do sjajnih literalnih priloga. možda bi posebno vredelo istaći te ško dostupnu doktorsku disertaciju nobelovca Ive Andrića pogotovo uzimajući u obzir i njanu trenutnu ak tuelnost.

Novembar je donso pregršt novih viceva. Teme su bile raznolike od Muje i Hase prako izbeglica dnavne politike i životinja do šaljivog leksike o predodredenom zanimanju. Novembar će ostati zapamćen po izbornma (organizovana su čak četiri zaosiaia VOTE e za vic mesece) a i po kadrovskoj promani u ovoj konferenciji – obzirom de je dosadešnji moderator Vlada Krstonošić ošišao na novu dužnosi (zvanu „moderator u resejanju") konferencije je dobila novog moderatora, Željka Krstića (*zkrstic*). Pobednički vic bio je na večitu temu „Cvrčak i Mrav" – malo je predug da b smo ga ovde prepričavali ali je opet mrav izvukao debji kraj. Evo zato prvoplasiranog vica iz drugog glasanja. Umro Bil Geles i dočao kod Svetog Petra. Ovaj ga pita da li bi više voleo da ida u Raj ili u Pakao. Bil naravno pita kako koj od njih izgleda. Sveti Petar mu pravno pokaže Raj – ljudi sede na jednoj sunčanoj poljani odmaraju se piju čaj, gledaju zalazak Sunca. A Pakao velike žurka treští muzika pije se

KAKO POSTATI ČLAN

Postupak učlanjenja počinje zvanjem Sezama – podesite parametre komunikacije na 8N1 (ako imate MNP mesem isključite MNP) i pozovite (011)648-883. Posle pozdravne poruke Sezam će ispitati prijavu:

Username:

Na vama je sama da izaberete NEW (novi korisnik) a Sezam će vas zatim pitati kroz dosta pristojnjavanja

Izbor imena

Jedan od prvih podataka koje treba da date je puno ime i prezime a polam i pseudonim pod kojim ćete ističiti ustupe Sezama. Jedinom uslovom pseudonim ostaje vaše pravo ime na Sezamu (ne može se menjati). Sa znači da se treba pažljivo izabrati - predlažemo da se još ne pomoć opredelite za pseudonim koji će vidite do poseda na vaše ime i prezime. Ukoliko nemate ideja, Sezam će vam predložiti pseudonim sastavljen od vašeg prezimena i jenog slava imena. Pseudonim može da ima između 4 i 16 znakova (preporučuju se 5-8 slova jedne više znakova znači vise kucanja, kako za vas tako i za one koji sa vama komuniciraju) i u njemu se mogu naći slova i znak tačka

Izbor lezinke

Sledeći korak je izbor lezinke (password) kojam štitite vaš račun – za razliku od pseudonima koji je javna informacija, lozinku znate samo vi i Sezam. Zato je izuzetno važno - naročito ubeagavajte da lozinka nisje jednaka pseudonimu ili da se sastoji od vašeg imena nadimka ili imena nekog bliskog srodstva – lozinka može imati između 5 i 8 znakova (slova ili brojeva, ili svega onoma čega je kombinacija). Ukoliko želite da promenite lozinku, to možete učiniti kasnije, ali ne možete promeniti pseudonim.

Izbor radnih parametara

Ostaje još da izaberete radne parametre (antiver protokol i slično – za bolje ako još ne znate šta je ka, pođite od Sezam uvek predlaženih podešavanja vrednosti)

Unos ličnih podataka

Pritkom unosa adrese i telefonskog broja budite veoma pažljivi jer netačne unesena adresa onemogućava kontakt Uprava Sezama sa vama, a samim tim i vaše učlanjenje. Unošenje ostalih ličnih podataka završena je prva faza vašeg prijavljivanja na Sezam

Redovno..

Sledećeg radnog dana će vam poslati pismo sa obaveštenjima o Sezamu i popunjenom uplatnicam. Ukoliko želite da proverite da li su podaci koje ste uneli ispravni pogledajte sledeći dana, unesite izabrani pseudonim i lozinku. Sezam će potvrditi da vam je pismo poslato te zatim će da dobijete podatke. Pošto obiljete pismo treba da zaberete period za koji se želi pačunati i za koji vam treba uplatite uplatite odgovarajući iznos

za 1 mesec	8 bodova
za 3 meseca	15 bodova
za 6 meseci	25 bodova
za 12 meseci	40 bodova

Vrednost bodo je vezana za kurs nemačke marke i manja se svakog ponedeljka u 8 časova ujutro. Ova vrednost će biti upisana na uplatnice koju dobijate od redakcije. Ako uplaćujete na svojim uplatnicama vrednost boda možete saznati se Sezama, ili ako se želite sjeviti glasom

Nakon izvršene uplate kompanju potvrde podaljite pismo na adresu „Mladun". Suševac vojvode Mišića 17, Beograd 11 faksom na telefon (011)649-140 ili (011)647-965. Uprava će vam ra dan prijema potvrde odrediti *pristup sistemu* a sledećeg dana će na vašu adresu im poslati Uputstvo za upotrebu Sezama, atraktivne štampana knjižica (100 stranaj pomoću koje ćete upoznati sve mogućnosti sistema (poštanci koji se preplate na mesec dana će dobijaju uputstvo dok ga još ne prodaje preplatu).

...i ekspresno učlanjenje

Oni kojima se žuri mogu postati korisnici Sezama za svega nekoliko sati. Javite se na njegan način a imes ćemo ih čekajući pi sme izvršite uplatu u poštu navedeni podatke. **Pečljalac** vaše ime prezime i pseudonim koje ste izabrali. **Primalac** sliCZ. Računan **Bvrha uplate** Prelplata na Sezam. Žire račun 60802-603-23264. **pezlv na broj** 35 108-5. Potvrdu o uplati pošalj te faksom do 13 časova (radnim danom) i već u popodnevnim časovima ste punopravni korisnik Sezama!

Dvostruko prijavljivanje

Ukoliko tokom predstavljanja imate probleme (npr veza ra prekida) pozovite ponovo i prijavite se navedeni kompletna podaka. Treba, ipak izvegavati višestruke prijavljivanje ja pamo i uplaćenu šaljeme same jednom. Prilnja ake ste se prijavili dva ili više puta kompletno ste korisničke ime koje je napisano na uplatnici

Koliko važi uplatnica

Cene naveiene u pismu važe 7 dana. – ukoliko ne izvršite uplatu u tom roku, mogućie su korekcije ali to ra znači da se morate pesove prijavljivati devolino je da pozivate Sezam predstavite se izaberem imenom i lezinkom i sistem će ispisati ažurne cene pretplate

Razgledanje sistema

Ukoliko žre prijavljivanja želite da razgledate sistem ra pe tanje **Username** odgovorite sa DEMO. Nači ćete se u demonstracionom režimu rada u kome možete da pregledate kompletna struktura sistema koja je identična „pravoj". Broj DEMO pozvanja nije ograničen a vreme za jedno sesnuj je 5 minuta

štampenjac gole žanske id. On naravno izebere Pa kao I – buć u kazan sa vrsilm ujem Kuke on kuke i vića da to nije ono što je video a na to će Đavo „Ono što si video je bio DEMO. I bar znaš šta je to"

Nastavak sa strane 13

„jugo 45" *WinTach* je brz jai korisi ekranska kate za prikaz slike

AUTOCAD: POTPUNA USKLAĐENOST

Superstation 3D je u potpunosti namenski proizvod. Projektovan je sa ciljem da se snaga grafičkih radnih stanica prenese na PC platforme. Osnovno polja primene ove kartice je dakle CAD/CAM, ali sva upućuje na to da je namenjena jednom jedinom softverskom proizvodu – AutoDesk CAD porodici *AutoCAD 3D Studio AutoShade* i *Animator Pro*. Korisnici ova tri paketa *Superstation 3D* obezbeduje preko 17 grafičkih formata – precizne crteže u razolucijama do 1280×1024 tačkica (256 boja) i prilodne boje do 16 7 miione nijansi (24 bitni *true color* pri maksimalnoj rezolucji od 768×576 tačkica) – uz brzinu grafičkih operacija koje je (prema podacima proizvođača) gotovo 10 puta veća od brzine standardne VGA grafike

Ovakva performanse *Superstation 3D* postiže za hvaljujući potpunoj usklađenosti sa popularnim AutoCAD paketima. Sa njima komunicira preko ADI (*Autodesk Device Interface*) drajvera koje je AutoDesk razvio za afikasan pogon periferijskih uređaja posebno grafičkih adaptera I TIGA grafičkog jezgra za komunikaciju sa TMS34020 procesolom ADI interfejs konsti dve lehnike ze brzo crtenje i osveževanje crteže dvostruku video memoriju (*double buffering*) i ekransku listu (*display list*). Kod tehnike „dvostruke memorije" crteži se formiraju naizmenično čas u jednoj čas u drugoj video memoriji a na ekranu se prikazuje sad rržaj one video memorje u kojoj je crtež golov To prekidno zneči da *Superstation 3D* ne crte pred očima konsnika već sa gotovi crteži pojavljuju trenutno prebacivanjem sa jedne na drugu video memoriju. Kod tehnike „ekranske liste" crtezi se skladiste u *programsku* memoriju grafičkog procesora kao niz proste čunatih grafičkih vektora i koordinata. Generisanje crteže iz ovako pripremljenih lista (koje su uzgled u verzji *AutoCad 12* u 32 bitnom formatu) je brže i lak še od stalnog obnavljanja CAD baze podataka o crtežu

Hercules ADI drajver obezbeduje konsnicima *AutoCAD* pakela i tri nove grafičke alatke – „pogled iz dalj ne" (*Bird's Eye View* – pčija perspektiva) za uda ljavanje „pogled iz bizine" (*Spyglass View* – dubnn) za približavanje slike i „meni sa prečicama" (*Shortcut menu*) „Pogled iz daljina" omogucuje da se u posebnom prozoru koji se otvare preko dela crteža koji se trenutno nalazi na ekranu prikaže komplatan crtež i nad njim pnmeoujuo pan i zum operacije bez posredovanja AutoCAD **display** menija. „Pogled iz bliz ne" otvere prozor za prikaz uvećanih detalja crteža na ekranu. Izbor se vrši pomerianjem posebnog kursora (*view box*) po crtežu a prikaz je trenutan (*real time*) Iz ovog prozora korisnik sa može ili vratu u tekući crtež ili tekući crtež zamaniti sadržajem prozore sa pogledom izblize. Sve ove operacije udaljavanje i približavanje slike leku glatko i izvršavaju se praktično trenutno. Pozornma se mogu manjati d menzije i mogu se pomerati po ekranu

Meni sa prečicama omogućuje da se česte komande veže za *pop-up* ikone i umesto zamornog prolaska kroz slabo sa menja-ma i komandama aktivirati jednim klkom na miš. Iastarnom ne tastatun ili dugmatom ne digitajzeru. Za ikone se mogu vezati jednostavne komande ali i skript programi na *AutoLISP*-u

Od brzih mašina nema mnogo konsh – ako nema ko da ih vezi *Windows* se kad se sve sabere i oduzme ovde pokazao kao prilično loš vozač. Ne vidimo zapravo kako *Superstation 3D* može da konkurišе ostalim grafičkim akceleratorima za ovaj sve popular nji operativni sistem S3 kartice (videl „Računera" 87) po ceni od 800 do 800 DEM donose poboljšanja u brzini od 2 do 3 puta sa manja neizvesnosh all i uz siromašniju paletu boja *AutoCAD* softver s druge strane konsti sva hardverske potencijala kartice

Ako projeklujete na računeru možda vam nedostaje upravo *Superstation 3D* de biste poboljšali svoju kreativnost I– produktivnost. Naša preporuka ako budete u prilici obavezno pogledajte a o kupovini odlučite sami. Nase upozorenje n pošlo ne kupujte naslepo pre nego što karticu proverite na delu sa onom aplikacijom zbog koje je i nabavljate. Cena od 7700 DEM je vredna svakog opreze

Korisna adresa

Micro Anime
11000 Beograd 29 novembra 1
Tel (011) 343-940

PRESTUP S PREDUMIŠLJAJEM

Dejan odgovara: Ja sam, za razliku od gospodina Tešića, koji je očigledno zgrožen mojim delom, i te kako uživao u njegovom pismu (članku?). Retko kad se srece tako lucidno objašnjenje nekih aspekta objektnog programiranja (krećući od teoretske podloge, dakako). No, uz to je g. Tešić naveo i niz mesta u mom tekstu za koja tvrdi da su netačna i nedorečena. Da sada za kaznu ne bih klečao na kukuruzu, dopustite mi da objasnim o čemu je reč:

Umesto da bacim na čitaoca sve što C++ ima da ponudi, što je prostup koji bi nesumnjivo zahtevao mnogo više prostore od jednog novinskog članka (nekoliko knjige bi bilo prikladnije), potrudio sam se da iz log obilja izdvojim one informacije koje bi čitaocu dale nekakvu osnovu u korišćenju objekata, i da ga „zagolicam", tako da po čitanju teksta pokuša da do kraja upozna C++. Odlicne kritike na Sezamu, kao i to sto je moj tekst završio kao #1 na tamošnjoj top-listi, samo su me učvrstili u uverenju da je ovakav prostup ispravan. Da ne bude zabune, niko se tamo nije jevio i rekao „Hej, tekst je super, eto, postao sam C++ programer", već su rekli da im je tekst razjasnio mnoge stvari koje im do sada nisu bile jasne, ta da će se sada ozbiljno pozabaviti C++-om. Čini se ispravno.

ZATVORITE ME S KNUTOM

Što se tiče „gradiva" koje sam za laj tekst izdvojio, ono se paradoksalno zasniva na Turbo Paskalu. Momci u Borlandu imaju zgodnu naviku da svoj paskal obogate sa 70 posto snage „onog drugog jezika" zadržavajući pri tom jednostavnost. Tako je TP u verziji 4.0 dobio unite „pozajmljene" iz Modula, u verziju 5.5 su stigli i objekti veoma slični onome kako ih C++ predstavlja, a tek je verzija 6.0 donela i **private** polja implementirana doduše, drugačije nego što to C++ radi. Kako ne znam nikoga ko ima problem sa shvatanjem objekata onako kako ih Turbo Paskal nudi, i ja sam se u svom tekstu „usudio" da predelavim samo klase, virtuelne funkcije, konstruktore/destruktore i naravno preklapanje operatora, koma nisam mogao da odolim. Za **friend** klase/funkcije i **protected** polja klase tu jednostavno nije bilo mesta, jeko su ove „opcije" veoma korisne. Ljudi, liški u objektnoj metodologiji pisanje programa najčešće ih zloupotrebljavaju. Čini mi se da bi bilo pametnije „sakriti" ih od novih programera, nego im deti nešto što u većini slučajeva ispadne kao tempirana bomba. Nije baš „information hiding", ali tu je negde.

„Omaške i nekorektnosti" iz mog teksta su tu (osim za „klasu kao sinlaksnu jedinicu") namerno. Smatrao sam da je mnogo bolje da kažem da klase „nije nišle drugo nego malo poboljšana struktura" nego da je citiram g. Tešića: „uz objekat osnovni koncept objektnog prilsupa teoriji sistema i kao takva je preuzeta i koristi se i u objektnim jezicima." Ajde! Ne vidim kako bi prosečan C programer koji nije ni prošao blizu teorije sistema mogao da shvati ovakvu rečenicu. Moj phstup je, iako možda netačan empirijski gledano, uspeo da objasni šta je to klase. Nekako se ne osećam loše zbog toga. Čak i Knut u predgovoru knjige o TeX u kaže da su „neke tvrdnje u knjizi materijalno pogrešne, međutim da to predstavlje način da konsnik shvati neke složene stvari na određenom nivou da bi, kada njegovo znanje napreduje, mogao da ih shvati u potpunosti!" Ako budemo robijali zbog naših gnusnih zlodela, nadam se da ćemo deliti ćeliju.

(Nagradna igra: Priznao sam da sam mislio jedno a pisao drugo. Ako uspete da dokažete da sam u isto vreme radio treće, moći ćete da se okušate i u novom kolu: „Ko je zapravo po znali policijer S.M.?")

SILOVANJE JEZIKA

Pa, to je to. Bez ikakve želje da dalje raspravljam o ovakav način diskusije u ostatku mog odgovora sledi dve tačke u kojima moram da iznesem svoje neslaganje sa g. Tešicem.

U jedanestoj lački on iznosi svoje (blago rečeno) neslaganje sa mojom konstatacijom da objekte ne treba gurati svuda. Da li je to tačno ili ne, to još niko ne zna, ali većina svetskih autoriteta na polju objektnog dizajna i klasifikacije (Booch, na primer) tvrdi da ipak *nije sve objekat*. Sidemo li sa Olimpa teoretske diskusije, stvari postaju još gore: pokušajte samo da preusmerite neki interapt na **member** funkciju (pravu, ne statičku) neke klase. Ili se upitajte da li na funkcijama najnižeg nivoa za recimo crtanje po ekranu, u kojima se svaka instrukcija broji, možde treba uštedeti na skrivenom **this** parametru. Ili se upitajte zašto i sam Stroustrup u svojim programima koristi **friend** funkcije (!) da bi prevazišao neka inherentna ograničenja objektnih jezika? Insistiranje na nemešanju objektne i strukturnane metodologije razvoja programa ne samo da je teoretski nepoželjno, nego je i praktično nemoguce! Uvek ću dati prednost objektnoj metodologiji kad mogu, ali insistiranje na njoj predstavlja bi pretvarje u silovanje samog jezika.

Usput, iako i meni ideja de neko od C++-a uzme samo najpotrebnije a da piše zapravo C programe zvuči loše, ne mogu se nikako složiti da je to gore nego da piše „suve" C programe. Nije teško zemisliti nekoliko klasa (string, vektor i matrica padaju na um) koje bi svaki C programer mogao i te keko dobro da iskonsli u svojim programima, bez ikakvog poznavanja objektnog programiranja. Velika je šteta ne iskonsliti sve što C++ nudi, ali, koliko god parče da odgrizete, nikada niste u minusu.

NA LEĐIMA C-a

U svom zaključku g. Tešić tvrdi da nije C++ u osnovi još uvek C. Moram da se i složim i ne složim (?) sa ovakvom konstatacijom. Reci se jedne strane da je C++ samo malo bolji C je kao kada bismo rekli da je Andrija Artuković „čovek koji je voleo da radi sa ljudima". Prednosti koje C++ donosi u odnosu na C su ogromne, radi se u stvari o kvalitativno novom jeziku. Ne žalosti, kada kažem da je C++ u osnovi još uvek C, to je zato što on ipak pati od mnogih ograničenja koja ima i sam C. Na primer, C++ ne dozvoljava da napravite operator ** za stepenovanje, a i , koje je takođe dobar kandidat ima problem sa prioritetima (^ slabije vezuje od operatora + pa lako izrazi ne bi bili „ono pravo"). Ili recimo klasa **LongLongint** ne može direktno od parsera da dobije broj

12435473127123572654724

već se dodele mora vršiti preko stringove

a = LongLongint ("12435473127123572654724")

Pelljanje sa parserom bi predstavljalo veliku muku, tako da je C++ uz izuzetak nekoliko novih operatora i ovde ipak napravljen na leđima C-a. To ne treba gurnuti ispod tepiha, već se treba u odnosu na to previlno postaviti pri kreiranju programa.

(Usput, moja konstantacija da je C++ još uvek C je podrazumevala da se mnoge stvari u C++-u rade kao i u C-u. Nisam želeo da objašnjavam u tekstu koncepte klase i polimorfizma, pa da me posle presretne neki čitalac sa pitanjem „Dobro, super su te klase, ali kako se u C++-u sabiraju brojevi?" Verujte mi, to se dešava.)

Da zapakujemo stvari, g. Tešić i ja se (ma koliko to čudno zvučí) slazemo da je C++ odličan izbor. Kao što to obično biva, g. Tešic će iz ove diskusije izaći uboden da sam ja potpuni lolirent i dilejs, ja ću misliti da je njega žena ončepila de mi napiše repliku zbog mojih šovinističkih primera, a vi ćete, ukoliko ste ovo preživeli, verovatno nastaviti da konslite ovaj prelepi jezik. Živi bili pa videli.

mikronova Software

Đure Jakšića 1a, 21000 Novi Sad, tel / fax: +38-21-26-251
26-275

PREDSTAVLJA SuitCASE

Ako želite:

Automatizovan razvoj aplikacije,
- Bavljenje projektovanjem a ne programiranjem,
- Aplikacije koje automatski rade i u mreži, (višekorisnički ambijent);
- Aplikacije koje su transakciono orijentisane (rollback i sl);

Ako NE želite:

- Da ste u začaranom krugu editovanje-kompajliranje-testiranje-debagiranje,
- Da ste rob a ne kreator Vaših aplikacija jer se najveći deo vremena bavite održavanjem aplikacija,
- Da Vam projektna dokumentacija leži u glavama projektanata;

APLIKATIVNI SOFTVER

- preko DVADESET kompletnih programskih paketa
- svi programi pisani u SuitCASE -u
- mogućnost vezane kupovine (aplikacija + SuitCASE)
- SAMOSTALNA dorada software-a (obj + lib + SuitCASE)
- dokumentacija i uputstva obezbeđeni

NOVELL mrežna oprema

- NETWARE 2.2, 3.11 i NETWARE LITE 1.1
- X.25 GATEWAY
- povezivanje udaljenih radnih stanica
- konsalting, servis

EXOS

Microdyne

- ultra brze ETHERNET kartice COAX (256/512 Kb RAMa, 80186 proces
- POCKET ETHERNET adapteri
- povezivanje sa HOST sistemima
- CDDI / FDDI (100 Mb/s)

- transiveri
- ripiteri
- bridževi
- pocket 3270 adapteri
- rešenja za olakšano kabliranje

TIPOGRAFIKA

NOVELL®

SYSTEM INTEGRATOR

BBS za podršku korisnicima SuitCASE i NOVELL: 021 / 26 - 275

ComTrad

the ART of computer making

ComTrad 386/40-105

Procesor 386/40MHz, 4MB RAM
Floppy uređaji 1.2MB i 1.44MB
Tvrdi disk 105MB IDE
Dva serijska i jedan paralelni port
SVGA grafički adapter 1024x768/1MB
SVGA kolor monitor
Mini tower kućište, ASCII tastatura, miš

2,795.-

Jo samo ComTrad može: poklon uz 386DX
i 486DX računare - ručni skener "A4 Scan"!

ComTrad YU • Gornji apartmani, V. Popovića 6, Beograd
Telefon: (011) 222-41-51, 222-26-51 • Fax: (011) 222-41-39
Radno vreme: 9-17, ponedeljak-petak

PRINT
ovi Sad
ovosodskog
ima 35
l. (021)
23-717
ax (021)
23-901

ComTrad Shop
Beograd
Cvijićeva 104
Tel. (011)
752-663
Fax (011)
752-663

INFOTRADE
Priština
Hekurudha 7
Tel (038) 25-830
Fax (038) 25-822

COMTRAD
KRAGUJEVAC
Kragujevac
Kragujevačkog
oktobra 102
Tel (034) 60-336
Fax (034) 67-117

MANIR
Rumo
Grobljanska 2
Tel (022) 421-265
Fax (022) 421-365

ComTrad
COMPUTERS

Dušan Popović

MS WORD 2.0

uputstvo za upotrebu

SPECIJALAN DODATAK ČASOPISA *računari*
februar 1993

Obrada teksta uz pomoć računara je evoluirala zajedno sa mogućnostima samih kompjutera. Nekada smo bili srećni da imamo kakvu takvu kontrolu nad tekstom koji obrađujemo. Na primer, *WordStar* je bio ideal programa za obradu teksta. Njegove mogućnosti su uzdizane do neverovatnih granica, vaše sada znamo da su one bile veoma skromne.

Programi koji su usledili *WordStar*-u imali su veće mogućnosti, ali su i dalje radili u tekst-modu. Tako još uvek nismo imali potpunu kontrolu izgleda teksta koji obrađujemo. Ono što je jedna od važnih mogućnosti – tekst se uglavnom i obrađuje da bi se na kraju odštampao, a šta znamo samo „otprilike" kako će izgledati, to jednostavno nije ono pravo. Veći tekst procesori su imali i veoma bitnu **Preview** opciju. Pokretanjem ove opcije tekst-processor je prelazio iz tekst-mode u grafički mod i prikazivao što je moguće vernije sliku dokumenta, stranice ili nekog njenog dela – onako kako će izgledati kada se odštampa. Time smo dobili delimičnu kontrolu izgleda dokumenta pre same štampe. I dalje je bilo potrebno pisati u tekst-modu, a za proveru izgleda povremeno pogledati *preview*, ali je to već bio veliki pomak.

Poslednja generacija tekst-procesora potpuno radi u grafičkom modu i pridržava se WYSIWYG („What You See Is What You Get" – „Šta vidiš na ekranu, dobijaš na štampaču") filozofije. Takav način rada je pre svega omogućila snaga računara, koja mnogostruko nadmašuje onu od pre nekoliko godina, a zatim i pojava grafičkih operativnih sistema kao što su *Windows* ili *OS/2*. Uz tekst-procesor koji se pridržava ovakvog načina rada stalna imamo kontrolu izgleda teksta, jer se svi elementi (podvučena i podebljana slova, razni tipovi slova, tabloe, slike) teksta vide upravo onako kako će biti i odštampani.

Grafički način rada programa povlači određene probleme – na primer, morate imati znatno jači računar koji će uspeti da osvežava grafički ekran onom brzinom kojom vi menjate tekst. Razlog za to je jednostavan: dok je u tekst modu dovoljno promeniti jedan bajt u memoriji računara za promenu slova na ekranu, za to vreme je u grafičkom modu potrebno promeniti x (što je veće, slovo, broj bajtova raste) bajtova za istu promenu. Drugim rečima, u koliko nemate dovoljno brz računar pisanje teksta u grafičkom tekst-procesoru će se pretvoriti u pravu muku – spor rad i produgo čekanje na odziv računara pri bilo kojoj operaciji sa tekstom ume spasmo da iznervira.

Sa druge strane, ako su tekstovi koje pišete pretežno grafički orijentisani (veliki broj tabela, formula i slika) gotovo da je neizbežna upotreba ovakvih tekst-procesora, jer bez njih kao da radite vezanih očiju.

Kao standard u grafičkom operativnom sistemu me za PC namenjenu se *Microsoft Windows* (porad nekoliko neuspelih pokušaja drugih firmi – na primer *Gem* "Digital Research"-a) — operativni sistem o kome sta imali prilika da više pročitate u jednom od prošlih umetaka. Standard nad standardima za obradu teksta pod *Windows*-ima postao je program iste firme – *Word for Windows* (tada u verziji 1.0). Posle njega se pojavilo još nekoliko tekst-procesora za *Windows* drugih proizvođača, ali ni jedan se njie ni približno slavi „Microsoft"-ovog programa. Sada je tu i novi *Word for Windows* u verziji 2.0 ispaliran i usavršen, tako da ima ogromne mogućnosti a lako se koristi.

Word for Windows je definitivno poslednja reč u oblasti procesora teksta. Pa, kada je već tako da vi dobijete šta nam nudi

TERMINI

Upoznajmo se najpre sa terminima koja su uobičajena u svoj oblasti rada se računarima. Objašnjenja nisu strogo tehničke definicije, već služe da upoznate početnicima približe neke pojmove.

TEKST PROCESOR je program za obradu teksta – program koji nam omogućava da sve zamisli „tekstualne prirode" stavimo na papir. Daleko od toga da su tekst-procesori (procesori reči, programi za obradu teksta) obična zamena za pisaće mašine – oni nam omogućavaju neohrabreno prepravljanje i menjanje teksta i prefinjenu kontrolu izgleda dokumenta PRE samog štampanja.

FONT je opis slova koja upotrebljavate pri pisanju tekstova. Postoje dve osnovne grupe fontova – bit-mapirani i vektorski. Osnovne razlike je to da se bitmap fontovi nalaze u obliku koji je lako menjati, čime se rešava i večiti problem naših slova, dok se vektorski fontovi nalaze u obliku vektora. Vektorske fontove je veome teško menjati, pa se ovi, potencijalno mnogo bolji fontovi, doskora nisu mnogo koristili. Prednosti vektorskih fontova su ogromne – dok kod bitmap fontova za svaku veličinu slova imamo posebnu datoteku na disku, kod vektorskih imamo samo jednu za sva veličine, a program sam iz jednog jednog opisa slova izračunava izgled slova u traženoj veličini. *Windows* od verzije *3.1* podržava vektorske (*True Type*) fontove, pa time i svi programi koji rade pod *Windows*-ima imaju mogućnost rada sa ovim fontovima. Tako, na primer, u samo jednoj datoteci (od nekih 70-ak kilobajta na disku) imamo sva veličine nekog fonta u rasponu od 4 do 127 tačaka. A taj raspon veličina je, verujte nam ne reč – više nego dovoljan.

DOKUMENT je svaki tekst koji pišete – od poslovnog pisma, preko novinskog članka, do romana. Svi vaši dokumenti koje pišete uz pomoć *Words*-a nalaze na disku i imaju nastavak DOC.

BOLD je način ispisa karaktera gde se ispisuju podebljana (masna, istaknuta) slova. Koristi se za i isticanje, na primer, ključne reči u rečniku.

ITALIK (italic) je pojam koji označava nakošena slova.

HEDER je tekst (obično kratak, jedan ili dva reda teksta) koji se pojavljuje u vrhu strane. Obično sadrži naziv – ime dokumenta, štampanje ili redni broj strane – koji treba da se nalazi na svakoj (ili svakoj drugoj, svakoj petoj, itd.) stranici.

FUTER je tekst sličan hederu, ali se nalazi na dnu svake strane.

ANOTACIJA su kratki komentari autora teksta. Služe kao neka vrsta podsetnika.

FUSNOTA se koristi za objašnjenje reči koja se javlja u tekstu. Određenoj reči se dodeli marker (broj u indeksu, slovo ili neki znak), a objašnjenje se ispisuje na dnu iste stranice.

POTREBNA KONFIGURACIJA

Već smo rekli da programi koji rade u grafičkom modu uvek zahteve od starijih tekst-procesora.

Za minimalan rad sa programom *Word for Windows* dovoljan je i AT sa 286 procesorom na 16 megaherca i sa 1 MB RAM memorije. Ove ahnatite bas apsolutni minimum i gledajte da ga izbegnete po svaku cenu – rad će biti jedva moguć. Osim toga, ako imate samo 1 MB memorije, mnoge mogućnosti programe će vam ostati nedostupne – na primer, svi dodatni programi (o kojima će biti reči kasnije).

Ako imate 386sx na 20 ili više megaherca, sa minimumom od 2 MB, rad već postaje solidan. Ekran se brže osvežava, a svi pomoćni programi se koriste bez problema. Naravno, svi računari brži od ove platforme poyednostavljuju rad i još brže izvršavaju sva operacije, ali tu nije prespor za ovaj program.

Hard-disk je posebna stavka. Sva *Windows* programu zahtevaju veliki prostor na hard-disku, ali je *Word for Windows* i u tome među prvima. U zavisnosti od toga da li instalirate ceo program ili samo neke delove, trebaće vam između četiri i petnest megabajta na hard-disku. Na ovu cifru dodajte još pet do deset megabajta za tekstove koje nameravate da pišete, pa saberite... Brzina hard-diska se bi trebalo da prelazi 25ms, pa to smatrajte apsolutnim maksimumom. Ako ste osuđeni na neki drajver za disk (*Stacker*[2]) koji se postavlja kao još jedna „sendvič" između hardvera na disku i procesora, primetićete koliko sporost diska utiče na sporost samog programa.

I na kraju – miš. Nećemo reći da je baš nezgodno vam (SVE su moža i sa tastature) ali gotove da je tako. Mnogo je lakše koristiti *toolbar* nego tastaturu, ma kako rad sa tastature bio podrđan

INSTALACIJA

Program se dobija na sedam AT disketa. Instalacija se vrši iz *Windows*-a, tako da je sam postupak donekle specifičan

- ❑ Ubacite prvu disketu u drajv i otkucate A.
- ❑ Otkucate WIN SETUP i instalacija počinje.

Dalje je potrebno odgovoriti na nekoliko pitanja – na primer potrebno je odrediti direktorijum u kojí će se program instalirati kao i koje delove programa želite na disku, a koje ne. Ako se prvi put sretete sa ovim programom, predlažemo vam da izaberete **FULL installation** opciju koja će instalirati ceo *Word for Windows*. Za to vam je potrebno malo vremena i petnaest (!) megabajta prostora na disku. Puna instalacija će vam pružiti i potpunu sliku o mogućnostima programa, a tada ćete lako izbrisati delove programa koji su vam nepotrebni

Program za instalaciju će sam prepoznati printere koji su instalirani i prema njima će instalirati i neke od mogućnosti (uglavnom ako radite pod *Windows-om 3.0*), pa je zato važno da printere instalirate za rad pod *Windows*-om PRE instalacije *Worda*. Ako radite pod *Windows-om 3.1* kaji kako već rekosmo, podržava vektorske *True Type* fontove, onda će automatski biti instalirano i nekoliko novih fontova, koji su potrebni pri radu sa editorom formula, delom programa u kome ćemo više reči kasnije

Tu dolazimo do prvog problema. Naime program koji instalira font koga koristi editor formule (*fences ttf*) ne uradi kompletan posao – font se, doduše, pojavljuje u pregledu fontova, ali se ne vidi na ekranu. Zbog toga je potrebno ovaj font deinstalirati iz programa **Control Panel** (opcija **Fonts**), pa ponovo instalirati. Ako je lole instaliran *fences* font (to je, u stvari, font koji sadrži razne oblike zagrada, a koje koristi editor formula) u formulama se umesto zagrade pojavljuju slova. O tome više malo kasnije.

Pri kraju, instalacija program pravi novu grupu u **Program manager** u i u okviru nje dve ikone – **setup** program i sam *Word* for Windows. **Setup** vam omogućava da naknadno instalirate ili dodajete delove *Worda* koji vam zatrebaju, ali ga sa diska možete i obrisati – sva naknadna instaliranja programa bez problema vrši i setup za diskete

OBAVEZNI KORACI

Za početak treba podesiti neke opcije *Worda* koje su neprilagođene našem tržištu. Objašnjenja koja sledi malo su preuranjena, jer još nismo objasnili nijedan postupak pod *Wordom* ali, ako pažljivo sledite uputstva, snaćete program podesen za početak rada. Postupak za ostala podešavanja opisan je u poglavlju „Podešavanje Worda"

- ❑ Prvo ćemo podesiti jedinice mere u *Word* u – podrazumevana mera je inć, a nama su potrebni centimetri. Za tu promenu je potrebno uraditi sledeće:
  - ○ Udite u **Options** deo **Tools** menija (Alt+O,O)
  - ○ Pritisnite jednom strelicu nadole da biste odabrali **General** podešavanje
  - ○ Pritiskajte TAB taster dok se ne aktivira polje **Measurement units.**
  - ○ Koristeći strelicu „dole" postavite selektor na **Centimeters.**
  - ○ Pritisnite ENTER da bi podešavanje bilo sapamćeno
- ❑ Ovaj korak je obavezan, jer bez njega editor formula neće ispravno raditi. Njime se deinstalira i ponovo instalira *fences* font.
  - ○ Startujte **Control Panel.** Startujte opciju **Fonts.**
  - ○ Označite font **Fences.**
  - ○ Kliknite na **Remove** i potvrdite sa OK.
  - ○ Ponovite **Add** opciju
  - ○ Izaberite **System** direktorijum i iz njega **Fences** font.
  - ○ Potvrdite instalaciju klikom na OK.

Ukoliko ste sve dobro uradili, trebalo bi da u spisku fontova ponovo vidite **Fences.** U protivnom će editor formula prijaviti nedostajući font.

- ❑ Ukoliko imate instalirane naše fontove u *Word* neće da ih prikaže u veličini od 10 tačaka (dakle ne često, na različitim video kartame) potrebno je deinstalirati *Times New Roman* i *MS Serif* fontove iz *Windowsa*, preko **Fonte** opcije **Control Panel** programa. Eve kako se to radi
  - ○ Startujte **Control Panel.**
  - ○ Startujte **Fonts** opciju
  - ○ Označite fontove *Times New Roman* (4 fonta) i *MS Serif*
  - ○ Pozovite **Remove** opciju i odgovorite sa OK na pitanja. Time će fontovi biti odjavljeni iz *Windows*-a, ali ne i uklonjeni sa diska.

Samom instalacijom fontova (i radu sa njima) često se detaljnije pozabavili u posebnom poglavlju, gde ćemo i ovaj problem njihovog (ne)prikazivanja rešiti na elegantnijn način. Dakle ovo rešenje je privremeno

- ❑ Poželjno je odmah podesiti veličinu strane na A4 format (koji se kod nas najčešće koristi), jer je podrazumevana veličina američki „letter standard" Podrazumevana veličine na smeta za pisanje teksta, ali smeta pri štampanju. Naime, program će „misliti" da je list papira u štampaču manji nego što jeste, pa će pogrešno složiti stranice. Veličinu papira često podesiti na sledeći način
  - ○ Pozovite **Page Setup** opciju iz **Favant** menija (Alt+T,U)
  - ○ Kliknite mišem na **Size and Orientation** dugme.
  - ○ Iz **Paper Size** dela dijaloga odaberite veličinu papira koju koristi vaš štampač (to je A4 format u najvećem broju slučajeva)
  - ○ Kliknite na **Use as Default** a zatim na pitanje odgovorite sa **Yes.**

IZGLED

Pretpostavićemo da već imate predznanje potrebno za rad pod *Windowsima*. Da biste naučili osnovne radnje sa ovim operativnim sistemom, preporučujemo vam da proučite umetak (iz „Računara" broj 88) koji se bavio samim operativnim sistemom. Prednosti rada pod *Windowsima* su višestruke – ne samo da možete raditi sa više programe istovremeno nego su i operacije dosta standardizovana, pa ako se naviknete na takve programe, učenje rada se bilo kojim novim programom je lakše –

Word for Windows startujete, naravno, tako što mišem kliknete na *Microsoft Word* ikonu. Naći ćete se u dosta popunjenom ekranu *Worda*, sa mnoštvom tastera i menija, ali nemojte da vas to zbuni – sve je tu zato da bi vama bilo lakše. Pa, da kreneno redom.

Ako ste prozor *Worda for Windows* proširili na ceo ekran (što vam preporučujemo), na vrhu ekrana će biti naslovna linija. Ako ste tek startovali *Word*, tu će pisati **Microsoft Word – Document1 doc.** To znači da ne morate ODMAH imenovati tekst koji upravo editujete – uostalom, možda ga i nećete snimati na disk nego samo odštampati, a tada vam ime i nije tako bitno. Tek pri prvom zahtevu za snimanje na disk program će primetiti da se radi o *default* imenu i ponuditi vam de ga promenite. Kako vidimo, podržana je vesma izvkana mogućnost *Word "Perfect* tekst-processora koju nazivaju „neimenovana dokumentima"

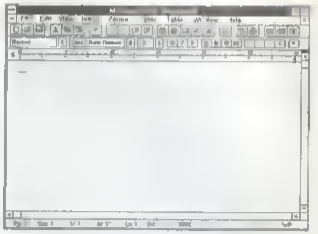
Ispod naslovne linije se nalazi standardni *Windows* meni. Opcijama je moguće pristupiti kako uz pomoć miša (koji se preporučuje) tako i se tastature. Opcija **File** (vidimo da je slovo **F** podvučeno) se poziva kombinacijom tastera **Alt+F**. Glavni meni ima sveje podmenije koji se koriste na isti način, i o kojima ćemo više reći kasnije.

Ispod klasičnog menija se nalazi *toolbar* linija. Na *toolbar* liniji su sličnim ikona koje nam služe da bez ulaska u meni za izbor opcije izaberemo neku od češće korišćenih mogućnosti „Microsoft" je pretpostavio prvi meni se ikonama, a onda je usledila čitava lavina programa koji koriste istu ideju.

Ispod *toolbar* linije se nalaze dodatne opcije za izbor načina pisanja – od stila i izbora fonta, do načina slaganje pasusa,

Četvrta linija je rezervisana za lenjir. Trenutna podela pri prvom startovanju *Words* je u inčima, dak su nama potrebni centimetri, pa ćemo malo kasnije naučiti kako se to podešava.

Ispod svega se (konačno?) nalazi „prazan papir" po kome se očekuje da počnemo sa pisanjem. Na samom dnu ekrana se nalazi još jedne linija koja nam daje neke statusna informacija – o tome na kojim koordinatama se nalazi kursor ili na kojoj strani teksta se trenutno nalazimo.. Svi delovi ekrana koje

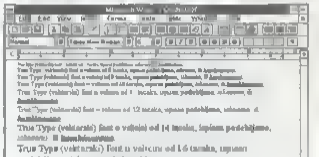


Početak pisanja. Prazna strana

smo nabrojali mogu se prilagoditi našim potrebama. Na primer, glavni meni je moguće potpuno prilagoditi, dodati opcije ili čak svoje makro-komande kao deleve menija. To vaši i za *toolbar* Neke deleve ekrana (kao što su toolbar ili lenjir) je moguće izbaciti, čime se povećava površina „papira" na ekranu, a samim tim znatno pogled na veći deo teksta odjed nom Koliko kontrolnih delova ćete ostaviti na ekranu, i kako će mengi izgledati zavisi od vas.

FONTOVI

Već smo govorili o fontovima koje koristi *Word* (i sam *Windows*). Naravno, i bit mapirani fontovi su podržani, ak vas korišćenje vektorskih (*True Type*) fontova je njihova veličina. Nije svejedno da li ste za jedan font u svira važnijim veličinama potrošili megabajt (često i više) ili samo stotinak kilobajta. Drugi razlog za korišćenje vektorskih fontova je principijalne prirode – ako kombinujete obe vrste fontova, pre ili kasnije ćete imati problema (oko snalaženja tipa „koji fajlovi predstavljaju ovaj font?" *Windows* koriste neke nedokumentovane postupke za rad sa vektorskim fontovima. Naime, stalno smo se pri instalaciji novog *true-type* fonta čudili kako taj font na ekranu na izgleda lepo, za razliku od fontova koje je *Microsoft* priložio uz *Windows* Razlog tome je mali broj tačaka na ekranu koje je moguće upotrebiti za „sliku jednog slova. Još čudnije je bilo što se pri radu sa veličinama od 10 tačaka, na ekranu ne vidi font koga koristimo nego neki od setaraških (uglavnom *Times New Roman*) Već smo vam dali jedan recept" kako se ta mana otkloniti, ali nismo objasnili i zašto je to tako, pa ćemo to sada učiniti Fontovi u *Windowsima* su svrstani u **familije** (na primer *Arial, Symbol*) Kada koristite neki vektorska font, *Windows* prvo proverava da li postoji instalirane biblioteke bitmap fontova ste familije (* FON) Na primer, uz *Windows* font *Arial* postoji FON biblioteka **arialb.fon** U njoj se nalaze bitmap opisi *arial* fonta u dve male i jako korišćene veličine – osam i deset tačaka (normal i bold) Sada



Kao u pravoj štampariji: Izbor fontova

je sve jasno – ako ste instalirali neki *Times* font, pe satim zahtevate rad se veličinum od deset tačaka, i ako imate naša slova u vektorskom opisu fonta, za rad se ekranom će biti ukorišćen bitmap opis koji se sadrži naša slova. Kako se ova mana može popraviti? Jedno od rešenja je promena imena familije fontova koju su problematični, što se može pomoću *Font Manger* editora vektorskih fontova. Drugo moguće rešenje je izmena bit mapiranih opisa, dakle FON datoteka. To je najlakše uraditi pomoću *Resource Workshop*-a koji se dobija uz neke od novijih *Borlandovih* C kompajlera. Na kraju, ukoliko vam

„kake" umesto naših slova po ekranu ne smetaju, nije ni previše važno – jake je sa pisanje po ekranu upotrebljeni FON datoteka, že štampanje se strogo koristi vektorski opis, pa se naša slova javljaju bez problema. **Font** se **instalira** tako što se pozove **Fonte** opcija *Control Panel*-a. Zatim se preke opcije **Add** izabera željeni font. Kada instalirate nove fontove, korisna je praksa smatiti ih u posebun direktorijum /na disku, a zatim ih instalirati sa **poništenom** (neščekiranom) **Copy fonts to Windows directory** opcijom Time izbegavate mešanje standardnih i dodatnih fontova, a olakšavate manipulaci je se njima

OSNOVNE OPERACIJE SA TEKSTOM

Kada startujemo *Word*, na delu koga smo nazva li „papirom" se može uočiti vertikalne crtica koja trepće ona u *Wordu* predstavlja kursor Već iz dos editora teksta znate kako se kursor pomiš. kako kucati neki tekst, slova se (jedno po jedno) pojavljuju na mestu kursora, a on se pomera za jedno mesto u desno Kada stigne do kraja reda, kursor prelazi u sledeći, a ako se istovremeno nalazi na dnu strane i cela strana se pomera na gore za jedan red.

Ukoliko ste do sada radili se nekim od programskih editera, verovatno ste stekli naviku da na kraju svakog reda (tamo gde procenite da je „otprilike" kraj) pritiskate ENTER, praćljevajući time kursor da pređe u sledeći red. To je bile potrebne zbog toga što editori programa operišu sa različito organizovanim podacima. Kod program-editora osnovni činilac teksta je bnija, dak je u tekst processorima osnovni pasus To znači de tekst kucate bez pritiskanja na ENTER do kraja pasusa, a program će se sam brinuti o prebacivanju višku teksta u sledeći red. Tak kade dodete do kraja pasusa: pritisnite ENTER. Ti ma stavljate de znanja tekst procesoru da želite novi pasus i tok tada kurser forsirate de pređe u sledeći red Velika je greška pritiskati ENTER na kraju svakog reda teksta, jer tada program svaki red sma tre posebnim pasusom (što nam nikako ne odgovara) I to nu stvara velike probleme oko slaganja teksta.

Tekst možete kucati na dva načina – u **insert** ili **overwrite** modu, odnosno novi tekst se može umetati u postojeći (na mestu gde postavite kursor) ili ispisivati preko postojećeg Znači, pritiscima na **Insert** taster se naizmenično menja mod u kome kucate Kada želimo da obrišemo deo teksta, možemo da to uradimo na dva načina – pomoću tastera **Delete** ili tastera na kome je nacrtane „strelica ulevo" i koji se obično nalazi iznad tastera **Entar**. **Delete** će obrisati slava koje se nalazi desno od kursora, a cačtak tekste se pomera ulevo Strelica ulevo će obrisati slovo levo od kursora, a tekst će se opet pomeriti ulevo. Kroz tekst se (osim već pomenutih tastera se strelicama) krećete i za više od jednog karaktera – koristeći **PageUP** i **PageDN** skačete za jedan ekran gore i dole

Ostale tastere koji se koriste za kretanje kroz tekst moćete videti u indeksu na kraju umetka. I ove osnovne operacije se tekstom su vam dovoljne da napišete neki kraći dokument – ručno pismo Pošto još nismo došli do slaganja teksta ne možemo previše utecati na izgled pasus, ali i te operacije ćemo naučiti uskore

KORIŠĆENJE MIŠA

Već smo rekli da korišćenje miša dosta olakšava rad se *Werdom* iako je moguće raditi i bez njega. Miš nam omogućava da neke operacije (koje se dostupne i preko menija, ali se nalaze u drugom ili trećem nivou pa je put do njih duži) pozovemo jednim klikom na *toolbar* – dakle veoma brzo. Svi programi koji rade pod *Windowsima* imaju jednu važnu osobinu – standardizovana komande To znači da se svi prozori kontrolišu na isti način, svi elementi se pozivaju i koriste na isti način Standardne su i sve operacije sa mišem Stavka menija se poziva klikom I ikone se *toolbar* linije se koriste na isti način – klik na određenu ikonu poziva akciju koju ta ikona predstavlja. Operacije se prosorima (povećavanje pozicionirane) verovatno već poznajete, jer su to postupci koji se laka nauče pri radu sa *Windowsima*. ali ćemo ih ponoviti.

- Prozor se povećava na oso ekran klikom na polje u gornjem desnem uglu prozora (polje se strelicom nagore) Kada je prozor povećan na ceo ekran, tada polje ima dva strelice a ponovan klik na njega vraća prozor na predašnju veličinu
- Prozor se smanjuje na ikonu klikom na polje poral prethodnog, na kome strelica pokazuje na dole.
- Pomeranje prozora pa ekranu vršimo mišem, tako što kliknemo na naslovnu linju, držimo taster na mišu, prevučemo okvir na željenu poziciju, i tak tada otpustimo taster
- Veličinu prozora menjamo tako što strelicu miša dovedemo na okvir prozora, strelica se pretvori u „dvostruku strelicu" o zatim pritisnemo taster na mišu držimo ga pritisnutog i pomerimo okvir na željenu poziciju pa tak tada otpustimo taster Miša koristimo i za brza kretanje po sadržaju prozora (e u našem slučaju po tekstu) klikom na određene delove **scroll bar** linije Kada pišete tekst u *Wordu*, imate na raspolaganju dve skrol linje – deono (vertikalan) i dole (horizontalnu) Tekst koji pišete možete da zamislite i kao veliki broj otkucanih papira složenih jedan i enad drugog Veličine teksta obično uveliko prelazi veličinu ekrane, odnosno količinu teksta koju se može videti ne ekranu Tada ekran služi kao prozor kroz koji gledamo deo teksta. Skrol linje

nam služe da se lakše orijentišemo gde se pronor nalazi u odnosu na početak i kraj teksta. Svaka skrol linija ima pet elemenata. dve strelice, dva polja između pokazivača položaje i sam pokazivač. Klik na bilo koji od ovih elemenata izvršava sledeću akciju

- Klik na bilo koju od strelica pomera tekst (u smeru strelice) za jedan red. Ovo je najmanji korak koji možemo preći mišem.
- Klik na polje između strelica i pokazivača položaja pomera tekst za jedan ekran – takođe u onom smeru u kojo pokazuje strelica bliža polju.
- Najveću slobodu kretanja po tekstu pruža sledeća postupak. kliknemo na pokazivač položaja (kvadratić na skrol linij), a zatim ne puštajuć taster na miši, pomerimo pokazivač na neko drugo mesto na skrol linji. Tuna možemo pomerati „prozor" po tekstu ukokavite, prelazeći brzo vahke kohćine teksta. Mi nam takođe olakšava rad sa blokovima teksta, ak će o teme biti više veći kasnje.

RAD SA DISKOM

Pošto ste otkucali svoj prvi tekst u *Wordu*, možete ga očitavpati i zaboraviti (ako ste ga unosili „za jednu upotrebu" odnosno ako vam je bio potreban samo za to štampanje) ili ga snimiti na disk i kasnje koristiti po potrebi. Takođe vaima važna mogućnost je snimenje teksta u nekom od formata koje podržava *Word*, koji je kasnje moguće uneti u neki drugi program *Word* ima mogućnost učitavanja tekstova koje ste sami snimili, ali i tekstova snimljenih u nekom drugom formatu – vrlo pogodno za prenošenje podataka iz programa u program Pri konvertovanju teksta treba na raspolaganju sledeće formate

- *Text Only* – običan tekst-fajl napravljen u bilo kom editoru.
- *DOS Text* – tekst se karakteriziua koji imaju kodove od 32 – 255
- *Rich Text Format* (RTF)
- *DOS text with layout* – tekst u kome program sam pokušava da prepozna pasuse i uvučene reči, i to sa dosta uspeha. Naravne, potrebna je pregledati tekst i ponegde korigovati odluke programa ali i ovakav rad je veliku pomoć.
- *Text with Layout* – još jedna varzija prethodne opcije.
- *Word Perfect* verzije 4 1, 4.2, 5.0, 5.1.
- *Word for Windows 1 0*
- *Word for DOS*
- *Word for Macintosh 4 0, 5 0*
- RTF DCA.
- *Excel* Worksheet – radna tabela iz *Excel*-a, programa za tabelarne preračune.
- *Dbase*
- *Lotus 1 2-3*
- *WordStar* verzije 3.3, 3 45, 5.0 ili 5.5
- *Windows Write.*

Snimanje teksta se poziva izborom **SAVE** opcije **FILE** menija ili pritiskom na **Shift+F12** Ukoliko je tekst nenumenovan (odnosno zove se „document1 doc"), program će zahtevati da unesete ime teksta, kao i opsuvine podatke o tekstu – ime autora, duža opis i slično... Ukoliko je program već snimao, (recimo nazvali ste ga PISMO, i *Word* nu je dodao nastavak DOC, pa se tekst zove PISMO DOC) ništa od podataka neće biti traženo a nova verzije teksta će zamenuti staru. Pre samog izbora imena, može se promeniti direktorijum u koji će tekst biti snimljen (po *default*-u je to direktorijum u koji je instaliran *Word*) ili **format** teksta. Obično je da se ova opcija ne koristi za snimanje teksta u nekom drugom formatu, jer je rad sa standardnim *Word* formatom teksta najbrži Druge formate treba koristiti samo pri konverziji finalne verzije teksta (ene verzije koju želita da obradite nekom programom koji zahtava poseben format) Još jedan način snimanja teksta je pomoću **Save** sa opcije iz **File** menija. **Save** sa opcija je slična običnom snimanju teksta, ali se od njega razlikuje po tomo što **uvek** zahteva unos imena. Ona se uglavnome i koristi za premenovanje dokumenta ili, što je još važnije, za konvertovanje u neki od formata koje podržava *Word* Već smo pomenuli kada ćemo koristiti koju od **Save** opcije – običnu **Save** opcuju kada redovne snimamo tekst, a **Save as** kada želimo da tekst konvertujemo u neki drugi format. Tekst se učitava pozivom opcije **Open** iz **File** menija ili pritiskom na **Ctrl+F12** Možete iza brati (na sasvim prirodan *Windows* način rada sa dijalozima) direktorijum i tekst koji želite da obradujete i učitati ga, ali ova opcija kroz ovnogo više mogućnosti Prva akrivena sugućnost je konvertovanje – ukoliko zahtevate učitavanje nekog teksta koji nije snimljen u *Word* formatu, program će prepoznati koji je ta format i ponuditi vam konvertovanje. Naravno procesa računara može biti pogrešan, pa je ostavljena mogućnost da zahtevate neki drug format teksta. Od toga kohko ste kovertera instalirali zavisi i koliko će vam stavki biti ponuđeno Vrlo korisna može biti i opcije **Find File** koja se takođe nalazi u okviru **Open** menija. Ovaj deo programa je bajc urađen čak i od „Norton"-ovog *FindFile-a*. **Find** će prvo pronaći sve dokumenta na disku, a zatim omogućiti nanoge korisne operacuje se njima pre samog otvaranja fajla. Recimo samo to da (u zavisnosti od toga šta smo odabrali) možemo videti snimgenu sliku nekog dokumenta ili ga otvoriti kao *read-only* To je izuzetne korisne opcija, jer možemo redom pogledati umnanjene slike dokumenata, koje nisu dovoljno velike da bismo pročitali bar ovo što piše, ali su dovoljne da se podsetimo „šta to beše" ili možemo videti njihova sumarne podatke– Ukoliko ste sigurni da želita da same pogledate neki tekst a na i da ga smrzate, nije loša praksa da koristite i **Read Only** opcuju Ona vam omogućava učitavanje dakumenta i sve operacije sa njim (štampanje ili isecanje delova potrebnih nekom drugom dokumentu) ali ne i snimanje – time se obezbeđujete od mogućnosti da slučajne promenute neki deo dokumenta (vi greškom otkucate nešto, a opcja automatskog cen manja teksta tu promenu učini trajnom)

TRI POGLEDA NA STVARI

U okviru **View** menija se kriju tri opcje, koje omogućuju različitu načine rada sa dokumentom **Normal** je način rada koji ćeta najčešće koristiti. Kada se on koristi, svi elementi dokumenta su prikazani onako kako će biti i odštampani – slike i formule su vide, fontovi su u željenoj veličini i deformacuji. Prelomi strana su naglašena isprekidanom horizontalnom linijom, a dokument se ponaša kao neprekidna traka papira. **Outline** je specijalan način rada. On omogućava da pregledame „kostur" dokumenta, i da operišemo sa različkovanim podacima o dokumentu. O čemu se radi? Kada izaberemo **outline** način rada, na *toolbar* linji (umesto lenjira) dobijemo još jednu linju sa tasterima. Na njoj se izdvajaju tasteri za brojevima od **1** do **9** i taster **All**. Još se nismo bavili operacijama sa stilevima, ali ćemo pretpostaviti da ćete se na ovaj deo objašnjenja vratiti kade vam **outline** mod bude bio potrebun. Kada izaberemo neki od brojeva, na ekranu će se videti same naslovi do tog nivoa. Na primer, izbor broja **2** će listati samo pasuse si stilovima **heading 1** i **heading 2**, dakle, dača nam pregled prva dva nivoa naslova. Ovo opcja je vrlo korisne kada pišete veći rad, pa želita da sredite naslove i ostale hijerarhijske elemente teksta. **Page Layout** uključuje pregled strana, tako da je izgled koji imate na ekranu najsličniji štampanom dokumentu. Sada su strane odvojene i fizički, a ne samo linijom. Kada je uključen ovaj režim rada, vide se i hederi i futeri, što nije slučaj u prethodne dva. Ovo je takođe i najsporiji režim rada, pa ćete ga koristiti ukoliko imate brz računar, ih ukoliko slažete dokument, pa radite finalne izmene. U okviru **View** menja postoji i opci je **Draft**. Njenim uključenjem rad postaje najbrži moguć ,jer se koristi jedan sistemski font za sve delove teksta, slike se ne prikazuju **Draft** je moguće koristiti paralelno sa sve tri načina prikaza teksta. Ukoliko ste prnsljeni da radite u ovom modu, niste daleko odmakli od tekst-procesora sa DOS.

HITNA POMOĆ

„Microsoft" je firma koja je poznata pe svojim help-sistemima. Ukoliko znate engleski jezik, help-sistem *Worda* će vam biti izuzetno koristan. Ma gde se nalazili, pritiskom na **F1** dobijata sarpnu dokumentaciju, i te u zavisnosti od trenutnog mseta na kome se nalazite u programu To znači da, ukoliko se nalazite na neku opcuji menja koje vam nje najjasnija, pritiskom na F1 dobjete opis te opcuje, a, ako se nalazite u nekom dijalogu, dobjete opise svih funkcya tog dijalopa i posebno same stavke na kojoj se nalazite Druga korisna stvar je povezanost – svako objašnjenje ime na samom kraju i **see also** deo, koji vam kazuje koje još delove podsetnika treba videti Ukoliko ste videli pomoć za neku opciju, ali vas zanimaju i dodatne informacuje, ova opcija vam omogućava da idete do željenog stepena složenosti objašnjenja. Ukoliko se ne nalazita na opciji za koju vam treba pomoć, nego vas zanima neka opšta operacja, povraćete opciju **Index** koja vam daje spisak, odnosno osnovnu i najopštiju podelu operacuja sa *Wordom* Dalje je moguće, sledeći povezane teme, doći do one tema koje vas zanima. Ali u okviru **Help** opcje glavnog menija krje se još jedna mogućnost – ukoliko želite da do kraja ovladate ovim kompleksnim tekst-procesorom imate na raspola ganju čitavu serju časova (*lessons*) razbitog stepene složenosti.

Ovi časovi su urađeni tako dobre da možete presnimiti da gledate crtani film, jer se sve operacuje odvijaju pred vašim očima i pri tome se koriste sve prednosti grafičke okoline Dok traje jedan „čas", program će vam prikazati kako za vrši neka operaci je (i to na više načina. ako treba), a zatim će od vas zatražiti da to ponovite Još samo nedostaje ocenjivanje na kraju godine... Još jednu opciju treba da spomenemo ukoliko ste do sada bili osuđeni (?) na korišćenje *WordPerfect*-a, prelazak na ovaj tekstprocesor će vam biti olakšan time što postoji i opcje **WordPerfect HELP** Kada startujete ovu pomoć, dobićete spisak operacija – sa jedne strane su *WordPerfect* operacje, a za druga identične operacije iz *Worda* Pri tome je najvažnije postojanje tastera **Automatic Keys**. Kada je selektovan ovaj taster (pe ste izabi iz *WordPerfect* helpa) pritisak na bilo kuju specifičnu *WordPerfect* kombinaciju tastera će vas vratiti u help-sistem, sa objašnjenjem za tu kombinaciju. Zgodna opcja, uz pomoć koje ćete za kratka vrema zaboraviti *WordPerfect* kombinacje i preći na standardna postupka rada sa *Wordom*

IZMENE U POSTOJEĆEM TEKSTU

Iza ovog naslova se krje sledeći korak u pisanju teksta – par složenijih operacuje za upravljanje tekstu, rad sa blokovima i pretraživanje.

U pisanju tekstava važi pravilo da je lakše uneti neki tekst (otkucati ga) nego ga posle menjati. Ima vše razloga za to – na primer, pri izmenama se mora voditi računa o više pojedinosti nego pri prostom pisanju– Zbog tih razloga, jedan zabiljan tekst-procesor mora da poseduje veliki broj opcuje koje olakšavaju izmene i prekrajanja teksta. *Word for Windows* nam omogućuje da poništimo poslednju akciju koju smo izvršli na tekstu – ovajedno da b mo obrisali slovo ili unetnuli sliku u tekst Ova mogućnost se može pozvati preko **Undo** —, opcije iz **Edit** menja, ih pritiskom na **Ctrl+Z**. Tekst iza rači undo se rasuja u zavinnosti od već pomnjana poslednje akcije Recimo da smo izbrisali deo teksta od kursora do početka reda (koristeći **Shift+HOME**,

Del kombinaciju tastera) Sada će u meniju pasti **Undo Edit Clear** dajući nam do znanja koja će izmena biti poništene pozivom undo opcije Naša je velika zamerka što **undo** funkcija nema više koraka, odnosno što ne pamti nekoliko poslednjih izmena nego samo jednu Tako da, ako ste izbrisali deo reda na način koji smo malo pre spomenuli, pa zatim greškom pritisli neki taster, možete se oprostiti od **undo** brisanja – sada se pritisak na taster smatra poslednjom akcijom **Redo** (F4) opcija iz **Edit** menija je suprotna prethodnoj – ona će poslednju akciju ponoviti još jednom Najlakše ćete efekat ove funkcije videti na primeru otkucajte neku reč i pritiskajte **F4** – svaki pritisak će ponovo ispisivati tu reč.

RAD SA BLOKOVIMA

Do sada smo se uglavnom bavili operacijama sa jednim slovom ili jednom rečju – umetanjem ili brisanjem Često se ukaže potreba da neku operaciju izvršimo na celom bloku teksta – rečenici, pasusu ili celom tekstu Korišćenje blokova teksta je najednostavniji i najbrži način za izvođenje nekih operacija se većoma količinom teksta. Kao primer ćemo uzeti jednostavnu operaciju za tekstorm – brisanja Već smo rekli koji su načini za brisanje slova levo ili na mestu kursora Ukoliko želimo da obrišemo veću količinu teksta (recimo jedan pasus) ovaj način brisanja postaje vrlo zamoran Sada obeležavanjem blokova teksta uveliko skraćuje rad – prvo obeležimo blok teksta koji želimo da izbrišemo a zatim pritisnemo DEL i blok je obrisan Blokovi nam još više pomažu kod operacija premeštanja velikih delove teksta, dakle, kod poslova slaganje i menjanja teksta, gde nam služe kao dobra zamena za „makaze i lepak" I, na kraju operacije se blokovima su možda najzastupljenije kod pisanja makro-komandi, što ćemo kasnije i videti. Blok teksta označavamo na više načina.

- **Jednu reč** označavamo tako što dvaput kliknemo mišem na nju
- **Jedan red teksta** označavamo tako što miša pomerimo na levu ivicu teksta (da je miš na pravom mestu znaćemo po tome što strelica nema smer gore-levo nego gore-desno) Kada smo postavili miša, kliknemo JEDNOM – čitava linija je označena.
- **Pasus** se označava isto kao i jedan red teksta, osim što ćemo kliknuti DVAPUT
- **Bilo koji deo teksta** možemo da označimo na dva načina – mišem ili preko tastature Mišem označavamo blok tako što kliknemo na mesto gde hoćemo da nam je jedan ugao bloka, i DRŽIMO taster na mišu Onda jednostavno „razvučemo" blok do drugog ugla i pustimo taster na mišu. Drugi način je pomoću tastature – dovedemo kursor na ugao bloke i držimo **SHIFT** taster Onda pomeramo kursor (pomoću tastera se strelicama) i pustimo SHIFT tek pošto smo obeležili ceo blok.
- **Ceo tekst** označavamo kao i kraju teksta, samo što pri tome držimo pritisnut **Ctrl** taster

Blok koji sam označih ostaje označen samo dok ponove ne počnemo da kucamo Znači, sve što želimo da radimo sa označenim blokom treba da uradimo me odmah, inače moramo ponove da ga označimo. Ako ste se navikli na druge tekst-procesore ili editore koji imaju mogućnost rada se blokovima primetićete da su ovde neke stvari drugačije urađene Kao prvo, do pojave *Windowsa* je bilo pravilo da označeni blok ostaje označen ma šta mi dokucavali (u n značeni blok tekste ih van njega), što u *Wordu* nije slučaj Drugo, ako ste označih veliki blok teksta, nr pošto ne počnjete kucanje novog teksta, jer će tada ceo blok biti izbrisan i zamenjao novim tekstom! Naravno kao spas za svakve greške nam služi **Ctrl+Z**, a, uostalom, na greškama se i uči Kada označimo blok teksta, za njim možemo izvoditi razne operacije. Blok možemo jednostavno izbrisati (što smo već videli), ili izvršiti neku operaciju sa *Clipboardom* Glavne opcije koje se bave blokovima teksta se nalaze u **Edit** meniju

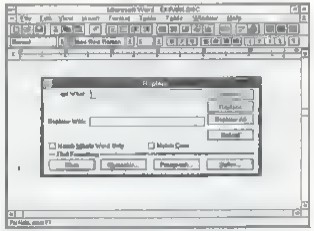
- **CUT (Ctrl+X)** će seći blok teksta. Ceo označeni blok će nestati iz teksta i biti upisan u *Clipboard*. O *Clipboard*-u možete više da saznate iz uputstva o *Windows*-u, ali podsetićemo vas da je to deo *Windowsa* koji nam omogućava da razmenjujemo podatke između programa. U ovom slučaju se *Clipboard* koristi za operacije u okviru jednog teksta, ali nas ništa ne sprečava da tako preuzet tekst upišemo u neki drugi program Ovu opciju ćemo koristiti ako nam je potrebno premeštanje teksta sa jednog na drugo mesto u okviru dokumenta.
- **COPY (Ctrl+C)** će prebaciti označeni tekst u *Clipboard*, ali će tekst ostati sačuvan Opcija je korisna ukoliko hoćemo da duplriramo blok teksta.
- **PASTE (Ctrl+V)** će prepisati sadržaj iz *Clipboard*-a na mesto gde se nalazi kursor u tekstu Primetićete da, kada želimo da blok teksta prebacimo na neko drugo mesto, moramo da
  - Označimo blok teksta.
  - Izvršimo **Cut** ili **Copy**, u zavisnosti od toga da li hoćemo da sačuvamo tekst na izvornoj lokaciji ili ne
  - Postavimo kursor tamo gde želimo da se pojavi prenesani tekst
  - Izvršimo **Paste** opciju.

Postoji još jedan način za pomeranje bloka teksta uz pomoć miša. Ukoliko smo označili blok, možemo ga prenesti na novo mesto tako što ćemo kliknuti mišem na blok teksta i jednostavno ga prevući na nove mesto u dokumentu Dok pomeramo blok na taj način, ispod strelica miše se pojavljuje kursor sastavljen od isprekidanih linija koji nam pokazuje tačno mesto gde će tekst biti umetnut Ukoliko u trenutku kada pustimo taster ne mišu ne držimo **Ctrl** taster, biće izvršena operacija **Move**

tekst nestaje na izvornoj lokaciji i pojavljuje se na određenoj Ukoliko držimo **Ctrl** taster, biće izvršen **Copy** - tekst će ostati na izvornoj lokaciji ali će biti i prekopiran na novo mesto Ovaj postupak je izuzetno praktičan – kada ga jednom naučite, nikada više nećete premeštati tekst uz pomoć menija

TRAŽENJE I ZAMENA TEKSTA

Traženju teksta često pribegavati – kada je u tekstu potrebno naći neku reč ili grupu reči Na ovu opciju se nadovezuje opcije zamene teksta – definiše se tekst koji za traži i tekst kojim se nađeno zamenjuje, a program obavlja ostatak posla. *Word* je ove opcije podosta razradio, tako da je sada moguće tražiti ne samo određeni tekst nego i određeni stil, paragraf ili način ispisa sleva. Opcije za traženje i izmenu teksta se nalaze u **Edit** meniju. Još da napomenemo ako ste izabrali blok teksta, a zatim pozvali opcije traženja ili zamene, one će pretraživati samo tekst u okviru bloka. U slučaju da blok ne postoji pretražuje se ceo tekst od kursora do kraja dokumenta, pa zatim (sa vašom dozvolom) od početka dokumente do kursora.



Zgodne za prepravke teksta. Traženja i zamena

- **FIND** će pronaći tekst koga ukucate u dijaloghoke Osim već standardnih opcija koje se posvjuju u svim programima (traženje unazad i unapred traženje samo cele reči, i traženje sa ili bez pravljenja razlike između velikih i malih slova), program može da traži po osobinama karaktera (same određen font, podvučen ili iskošen

Nadi ili zameni	Znak	Objašnjenje
Bilo kaj karakter	?	Na primer AC? će naći „aca", „accept" i „peace" Ovaj znak ne može da se nađe u tekstu za zamenu
Znak pitanja (?)	*?	Ako tražite **Kako*?** će naći „Kako?"
„Beli razmak"	*W	Bilo kakva kombinacija razmaka, tabova EOL karaktera. Ne može se navesti u tekstu za zamenu
Tab	*t	Mora za kontatiti malo t.Kraj paragrafa
	*p	Takođe, malo p.
Kraj linije (EOL)	*n	Malo n.
Kraj strane	*d	Malo d.
Kraj sekcije	*d	Kraj strane i sekcije se nalaze istom komandom.
Duplira nađeni tekst	*m	Na primer ako unesemo „unari" u Find What dijalogu i „Reč*m" u replace dijalogu dobijamo „Računari" Može se koristitsaino u *replace* delu dijaloga.
Sadržaj Clipboarda	*c	Koristi se samo u dijalogu „šta zameniti"
Neprekidni razmak	*s	Specifična vrsta razmaka, mesta na kome se ne može podeliti red.
Opciona crta	*-	Dosta egzotična opcija, retko potrebna.
Shift+6 znak	*^	Naravno, niste zaboravili da se ova kuca kao Shift+6, Shift+6?
Ascii karakter	*nnn	Ovo se zamenjuje slovom čiji je decimalni broj nnn
Ansi karakter	*0nnn	Ansi i ascii karakter set se razlikuju pre svega u raporedu pojedinih slova u karakter setu Ova opcija je identična ascii karakteru ali se radi o ansiju

Rad sa opcijom za zamenu teksta ćemo videti kroz nekoliko primera.

- Hoćemo da zamenimo sve pojave reči Microsoft rečju Borland
  - Pozovemo **Replace** opciju **Edit** menja.
  - Na **Find What** pitanje odgovorimo sa *Microsoft*, a na **Replace With** se *Borland*
  - Kliknemo na **Replace All** dugme.
  - Ukoliko kursor nije bio na samom početku teksta, *Word* će nas pitati da li želimo da zamenimo ceo tekst ili samo od kursora nadalje

- Hoćemo da sve **bold** karaktere u tekstu zamenimo **italic** karakterima.
  - Pozovemo **Replace** opciju **Edit** menija.
  - Na **Find What** pitanje odgovorimo sa ?, a na **Replace With** sa "*n (kuca se kao Shift+6,n)
  - Dok je kursor u okviru **Find What** pitanja, kliknemo na **Character** dugme.
  - U dijalogu koji dobijemo kliknemo na **bold**.
  - Postavimo kursor na **Replace With** pitanje, pa kliknemo na **Italic** opciju iz **Character** menija.
  - Klik na **Replace All** i svi podebljani karakteri u tekstu su sada iskošeni

- Isti zadatak kao prethodni, osim što bold karaktere sada menjamo *heading 1* stilom.
  - Svi koraci osim ovog su isti kao u prethodnom slučaju. Sada (dok je kursor na **Replace With** pitanju, kliknemo na **Styles** dugme. Iz dobijenog dijalog-boksa zatim izaberemo željeni stil (u našem slučaju heading 1).-O stilovima malo kasnije, ali je ovo dobar primer operacije koja se često primenjuje u *Wordu*. Sve dalje operacije zamene delova teksta su varijacije ovih primera. U svakom slučaju, ove funkcije su vam dobre uradene i lako se uče, a opak omogućavaju punu slobodu izmena (svi za sve?)

SLAGANJE TEKSTA

Slaganje teksta je širok pojmo – uglavnom označava podešavanje izgleda teksta pred štampanje. Naravno različiti poslovi imaju različite standarde, pa je za jednu primenu dovoljno koristi tri načine i spise slova (normalna slova, iskošena i podebljana), a za drugu je potrebno dobre se pomučiti da bi se sve tabbasa, slike i formule lepo uklopile. Prvo ćemo obraditi neke jednostavne operacije – zadavanje načina spise teksta koga kucamo ili bloku teksta.

Izbor fonta se vrši kliknem na *toolbar* ili preko menija **Ctrl+F**. Ovo je veoma važna opcija za definisanje izgleda dokumenta. Standardni *Windows* fontovi, koje *Word* podrazumeva sinovnim za početak rada, imaju jednu veliku manu – nemaju naše slova. Ako ste nabavili neki font koji ima naša slova (i pravilno ga instalirali pod *Windowsima*), prvi potez pri radu će biti promena fonta. **Veličina fonta** se bira pritiskom na **Ctrl+P** ili klikom na *toolbar* (izbor veličina fonta zavisi od samog fonta – ukoliko se radi o *True Type* fontovima (a mi stalno preporučujemo rad sa njima), raspon je više nego dovoljan, od 4 do 127 tačaka. Već smo govorili o **bold** (podebljanom) ili **italic** (iskošenom) načinu spisa slova. Kada četa koji spise koristiti zavisi uglavnom od ukusa i nekih pravila koja usvojite. Recimo, **bold** slova možete koristiti kada želite da istaknete neku ključnu reč u rečenici, a **italic** kada navodite ime neke firme. Način spise se zadaje na jednostavan način – preko tastature (što je i najbrži način) ili klikom na *toolbar* **Bold** način spisa uključujemo pritiskom na **Ctrl+B** ili klikom na ikonu se slovom **B** na *toolbar* linij. U oba slučaja ikone menja boju (da juči nam do znanja de je tekst na kome se trenutno nalazimo upisan **bold** načinom), a sev takst koji za tim kucamo biva upisan na ovaj način. **Bold** spise se uključuje ponovnim pritiskom na **Ctrl+B**. Ovaj način upisa se može zadati i tako da već otkucan tekst dobije **bold** upis naknadno. Rad se na sledeći na čin prvo obeležimo blok teksta koji treba da bude i spisan podebljano, a zatim pozovemo **bold** opciju i dobijemo željeni rezultat. **Italic** upis se odreduje na gotovo identičan način. Sada je potrebno pritisnuti **Ctrl+I** ili kliknuti na ikonu sa slovom **I**. Sve operacije su potpuno iste uključujući i one se rad se blokovima teksta. **Underlined** (podvučen) upis čete verovatno reda koristiti od prethodna dva. I on se koristi na standardan način a poziva se pritiskom **na Ctrl+U**, ili klikom na ikonu se slovom *U*. Tekst možemo pisati i u **Superskript** (izdignuta slova, kao kada pišemo eksponente) ili u **Subskript** (slova se spuštena u odnosu na osnovnu liniju teksta) režimu, pritiskom **na Ctrl+Shift+=** i na **Ctrl+=**. Sve načine upisa možete kombinovati po želji (na primer subskript+bold+italic), bez razmišljanja da li i mate takav font ili ne (kao što je to slučaj kod tekst procesora koji rade pod DOS-em). *Windows 3.1* je taj koji se brine da sam izračunat željenu veličinu vektorskog fonta, i da na njemu izvrši potrebnu deformaciju.

Veoma važan deo slaganja teksta je uravnavanje pasusa. Pasus može biti uravnat ulevo, udesno, po obe margine ili centriran. Ovakvim određivanjem izgleda pasusa već zaključno o izbijnje vrste slaganja dokumenata, jer lepo određeno uravnanje mnogo utiče na kompletan vizuelni utisak. Osnovno uravnanje (sa njim se počinje rad) je **uravnanje po levoj margini**. Pasus složen po levoj margini da i mala savršeno ravnu levu ivicu i verovatno nazubljenu desnu. Uključuje se **Ctrl+L** kombinacijom tastera. Ukoliko je obeležen blok teksta koji obuhvata više pasusa, uravnanje važi za sve pasuse. Suprotno lavom je, naravno, **desno uravnanje**. Startuje se pritiskom na **Ctrl+R**. Sada je desna ivi se složena „pod konac", a leve ne. **Centriran ispis** je koristan kod slaganja naslova. On će bez obzira na broj slova u redu složiti pasus taka da bude savršeno centriran. Tastern za poziv ove funkcije su **Ctrl+E**. **Uravnanje po obe margine** je najkompleksniji tip uravnanja. Ono teoretski obezbeduje da se obe margine savršeno ravna. Mana ovog načina slaganja je da se zbog savršeno složenih margina žrtvuje razmak reči u redu. O čemu se radi? Program će prve u jedan red smestiti onoliko reči koliko bi smestio uravnavajući po levoj margini, a prazan ostatak reda (koji može biti manji ili veći) će popuniti tako što će reči **razmeleti** ave dok tekst ne dodirne desnu marginu. Kolike će ova operacija biti kvalitetna, zavisi od više faktora, ne primer od drajvera za printer koji je instaliran za rad pod *Windowsima* ili od veličine fonta i dužine reči u tekstu.. Naš savet je da, ukoliko slažete neki dokument (na primer pismo) obavezno pokušate da ga složite koristeći ova uravnanja, pa tak ako su rezultati nezadovoljavajući da pokušate neko od prostijih uravnanja. I, na kraju, tekst je moguće složiti i više stubaca, tako da liči na novinski članak. Broj stubaca vežu za ceo tekst ili za obeležem blok teksta. Određuje se na sledeći način pozovete **Columns** opciju iz **Format** menija (**Alt+T,O**)..e zatim odaberete broj stubaca. Iz istog dijaloga se mogu popdesiti još neke osobine stubaca. Na primer, između njih može a ne mora da postoji linija koje ih razdvaja, međurazmak je moguće podesiti.. Broj stubaca se može takode odrediti i klikom na *toolbar* – na ikonu koja omogućava interaktivan (pomeranjem miša) izbor

RAD SA STILOVIMA

Da ne bismo svaki put ponovo birali kombinacije koje su nam potrebne (recimo podebljano + iskošeno), one se mogu složiti u **stilove**. Spisak svih raspoloživih stilova se može videti u linju ispod *toolbara* sasvim lave opcije. Klikom na strelicu koja pokazuje nadole (nalazi se odmah pored trenutno aktivnog stila), dobijamo spisak ostalih stilova, pa jednostavno izaberemo onaj koji nas zanima. Sve što nadalje pišema će se ispesvati odabranim stilom. Korišćenje stilova nas ne sprečava da i dalje, kada nam te zatreba, tekst složimo tako što ćemo ga „ručno" definisati. Na primer, ako želimo bold (podebljana) slova da bi istakli neku reč, ne moramo definisati takav stil – jednostavno kliknemo **na B** ispod *toolbara* ili pritisnemo **Ctrl+B** i dalje kucamo se podebljanim slovima. Ako ste ranje radili sa *Xerox Ventura Publisher*-om, najednostavnje je da vam kažemo da je stil u *Wordu* isto što i tag u *Ventura*. Recimo da želimo da definišemo stil koji koristi **bold** + **centriran** ispis Stil se definiše na sledeći način

- Pozove se **Style** opcija **Format** menja, ili se pritisne **Ctrl+S** kombinacija tastera.
- Unese se ime novog stila ili se odabere nekoliga želimo da promenimo iz spiska već postojećih.
- Kliknema mišem na **Define** i dobijemo dosta oblman dijalog boks.
- Sada biramo elemente koji nas enaimaju – u našem slučaju kliknemo na **Character** pa izaberemo bold ispis slova. Zatim izaberemo i **Alignment Left** iz **Paragraph** dela dijaloga Sve vreme u „descripiton" prozoru vidimo osobi na koja sme dodali stilu (u našem slučaju Left+Bold)
- Kada smo definisali novi stil, kliknemo **na Add** te **template** pa na **Apply** čimo stil ostaje za pamćen kao deo aktivnog templejta. Stil može da se hazira i na nekom već postojećem. Na primer, mje redak slušaj da opis stila glasi **Normal+14pt+**,.., pri čemu taj stil ima sve osobine Normal stila, plus nekoliko dodatnih

Pošto u *Wordu* možemo pisati različite stvari, tekstove gde se koriste potpuno različiti stilovi (na primer novinski članak i memorandum) broj stilova seže dosta da poraste, a tada bi oni postali svoja su prutnost – desile bi se da stil koji nam je potreban tražimo među gomilom nepotrebnih. Zato se i stilovi mogu svrstati u grupe.

Templates, su šabloni ili, u slobodnom prevodu, skup stilova. Sva stilove koji se koriste pri pisanju iste vrste dokumenata svrstavamo u jedan templejt. Kasnija, kada počinjemo pisanje novog dokumenta, jednostavno odaberemo templejt koji nam je potrebon, a tada su nam svi potrebni stilovi dostupni **Templates** omogućava i mnogo više od toga. Dovoljno je da jednoso podesimo vrednosti strana, orjantacuju (vertikalna ili horizontalna strana), margine i sve ostale parametre papirja, a zatim to snimimo kao novi templejt. Zamislite, recimo da smo kao jedan tempiejt definisali A5 stranu a kao drug A4. Kada startujemo nov dokument kao A5 templejt, automatski imamo podešene sve parametre potrebne za rad. Kompletna verzija *Word*-a nudi dvadesetak već definisanih templejta. Oni, na žalost, odgovaraju američkim pravilima pisanje a na našim, pa mogu poslužiti samo kao ideje za vaše stilove Znači, više za gledanje nego za korišćenje **Novi templejt** se definiše na sledeći način

- Pozovete **New** opciju **File** menja.
- Kliknete na **Template** pa zatim na **OK**. Ako ste ništa posebno naglasili, novi templejt će biti baziran na normal templejtu.
- Dobili ste prazan templejt Sada je potrebno definisati sve potrebne parametre – veličinu strane, margina. stilove, standardni tekst, pa čak i makro-naredbe koje će važiti samo u tom templejtu.
- Kada ste definisali sve što ste želeli (uglavnom stilove), pozovite **Close** opciju **File** menuja, i odgovorite potvrdno na pitanje

HEDERI I FUTERI

Heder je „tekst iznad teksta", odnosno kratak tekst koji se pojavljuje na početku svake strane Obično se sastoji od 1-2 reda i sadrži neki od elemenata za koje je uobičajeno da se pojavljuju na svakoj strani (broj strana. ime knjige ili štampariye) **Futer** je identičan hederu, osim što se nalazi na dnu strane. U heder i futer možemo direktno upisivati sadržaj (polja, tekst) ukoliko se nalazimo u **page layout** modu (ili nam je potrebna veća preciznost sadržaje hedera i futera) pozivačemo **Header/Footer** opciju **View** menija. Nikako nam nije jasno zašto se ova opcija

nalazi u View meniju ali nije problem navići se i na tu nedoslednost. U dijalogu koji dobijamo pozivom ove opcije, imamo mogućnost definisanja hedera i futera, ali i neke veoma korisne dodatne mogućnosti – mogućnost razlikovanja parnih i neparnih strana ili razlikovanje prve i ostalih strana. Koristi ove opcije ćete primetiti kada kao prvu stranu dokumenta definišete naslovnu stranu (na kojoj heder i futer ni su potrebni), a zatim jednostavno „pomerite" hedere na sledeću stranu.

UMETANJE ELEMENATA...

Word for Windows 2.0 raspolaže velikim brojem elemenata koje je povremeno potrebno ubaciti u tekst. **Insert** meni objedinjuje sve elemente (osim tablica) koji se na raspolaganju. Do sada nismo našli nešto što nam je bilo potrebno, a nije obuhvaćeno preko ova dva menija. Što je još važnije u **insert** meniju se nalazi i opcija **Insert object**, što nam omogućava da u tekst ubacimo neki od već postojećih objekata (iz drugih programa – slike, grafikoni i tabelstov). Prvo ćemo kratko pogledati manje složene elemente:

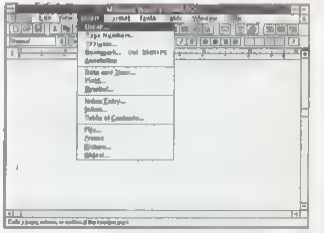
Kraj strane umećemo u tekst izborom **BREAK** opcije **Insert** menija ili pritiskom na **Ctrl+ENTER**. Ova opcija služi da se na željenom mestu označi kraj stranice. Zamislite, recimo, da složite naslovnu stranu nekog teksta. Pošto ste je složili (a da počeške sledeće ima još deste mesta) jednostavno umetnete **break** i .. nova strana je tu

Numerisanja strana vršimo pozivom **PAGE NUMBERS** opcije **Insert** menija. Ova komanda omogućava veliku slobodu pri izboru načina numerac̆e, gde se broj strane može centrirati po želji, u hederu ili futeru. Ovu komandu ćete koristiti kao najlakši ali ne i jedini način za numerisanje strana – kompleksniju numeraciju strana izvodimo ručno, pomoću polja.

Fusnotu umećemo pozivom **FOOTNOTE** opcije **Insert** menija. Postoje više načine za označavanje fusnota. Označu za fusnotu možete odabrati sami ili prepustiti računaru da vodi evidenciju o fusnotama. Sta god da ste odabrali, dalji rad je identičan: znaka fusnote se pojavljuje kao indeks iznad reči koju objašnjava, a vi ukucavate samo objašnjenje (komentar) u dno sa fusnote. Pošte kliknemo na **Close**, fusnota je zapamćena. Fusnote koje smo ranije uneli možemo u svakom trenutku da menjamo. Računar sam vodi računa o mestu fusnote, odnosno, ako reč za koju je fusnota vezana prade ne neku drugu stranu (usled pomeranja ili umetanja teksta) i same fusnota će se prebaciti tamo. Koliko je opcija dobro rešena videćete čim počnete da slažete neki tekst uz pomoć *Word-a*.

Marker pozivamo preko **BOOKMARK** opcije **Insert** menija, ili pritiskom na **Ctrl+Shift+F6**. Ovi markeri su zamišljeni ne samo kao oznake u tekstu nego služe i za generisanje kros-referenci

Poziv **ANNOTATION** opcije **Insert** menija umeće **anotaciju** u dokument. Anotacija je kratak



Umetanja elemenata

komentar ili podsetnik za autora teksta i pri štampanju ih pregledu se ne pojavljuje kao deo teksta.

DATE AND TIME opcija **Insert** menija nam služi da umetnemo datum ih vreme na mesto kursora. U stvari, na to mesto se umeće polje za vreme, a da bismo dobili naš format datuma morate ga podesiti u **Control Panel** u Posle tog podešavanja, format koji nam odgovara naći ćo se na prvom mestu ponuđenih formata.

SYMBOL opcije **Insert** menija liči na slično opciju u paketa *Windows 3.1* Koristi se kada je neki karakter nemoguće dobiti direktno sa tastature (dobro, ne baš nemoguće, ali ko bi još pamtio sline Alt+nnn kodove slova) Dobija se pregled svih „alternativnih" aktivnog fonta, a izbor slova koja želimo da umećemo se vrši uz pomoć miša ili sa tastature **INDEX ENTRY** vam omogućava da unesete opis koji će se pojaviti kada budete pravili tablicu indeksa. Ovaj tekst će biti sačuvan kao skriveni, i pojaviće se samo u tablici.

INDEX opcija **Insert** menija će sakupiti sve indekserane pojmove i složiti ih po abecedi–tako da uz svaki broj strane (ili strana) na kojima se može naći. Ova tablica je neophodna u slučaju da pišete i ole složeniji rad.

Sadržaj se dobija pozivom **TABLE OF CONTENTS** opcije **Insert** menija. Već smo rekli da je moguće sve delove teksta snabdeti raznim stilovima. Sada smo dobili još jedne (nezetno korisne) upotrebe stilova. U okviru **normal** templeta već postoje četiri definisana stila – tri za naslove i jedan za tekst Ako glavnim naslovima dodelimo stil **heading1**, podnaslovima stil **heading2** i međunaslovima **heading3**, *Word* sam izvlači sadržaj dokumenta. Sadržaj će biti ispisan tako da su pod naslovi (heading2) uvučeni u odnosu na glavne naslova, e apet, međunaslovi u odnosu na podnaslove.. Ono što dobijamo na ekranu je veoma lepo raspoređena tablica sadržaja, se brojem strana na kojoj se svaki naslov nalazi

Ceo dokument najlakše umećemo pozivom **FILE** opcije **Insert** menija. Ukoliko ne želite da umećete ceo tekst (ili se on već nalazi kao „otvoren") bolje je koristiti rad sa blokovima i **Clipboard**-om.

Poziv **OBJECT** opcije **Insert** menija je mogućnost koju smo već pominjali – povezivanje programa preko objekata. Kada pozovete ovu opciju dobijete spisak objekata koji su na raspolaganju Primećićete objekte koje je instalirao sam *Word (Equation Editor Microsoft Drawing Microsoft Graph i Microsoft Word Art)*, ali i neka koji su „prozvoli" samih *Windows (Paintbrush picture* ili *Sound* na primer...) Aplikacije koje kreiraju ove objekte su izuzetno korisne i zato ćemo im kasnije posvetiti posebnu pažnju. Pošto sta kreirali neki objekt preko ovog menija, obeležavate ga jednim klikom miša (pa ga sada možete pomerati ili manipulisati njime na drugu načine), a se dva klika mišem pozivate aplikaciju kojom je taj objekt kreiran

RAD SA SLIKAMA

Veoma često ćete u toku pisanja nekog dokumenta imati potrebu da umetnete sliku u tekst Cak se može reći da je to najčešća od svih operacija umetanje elemenata. Pošto je posao slaganja teksta i slika prava mala umetnost, a kako je *Word* dobro podržan (i olakšao) rad sa slikama, ni čemu se potrudili da ove postupke što bolje objasnimo. Sliku umećemo u tekst na mesto kursora pozivom **PICTURE** opcije **Insert** menija. Slika će biti umetnuta u tekst ili u ram, zavisno od mesta kursora. Ukoliko je slika dovoljno mala, može umetatu tekstu da se slobodno smesti oko nje. Ukoliko je veća, možda je bolje otvoriti okvir pa ubaciti sliku u njega. Podržan je veliki broj formata, i bit-mapiranih i vektorskih slika. Preporučujemo da se kad god je to moguće trudite da u dokument uvozite vektorske a ne bit-mapirane slike, jer, kada dođe do promene veličine (a retko kad slika „iz prve" odgovara mestu na papiru), vektorske slike to mnogo bolje podnose Delava se da bit-mapirane slike koje smo snimili ili povećali izgube dosta detalja i oštrine. Podržani su sledeći formati slika.

- *Windows Bitmaps (* bmp)*
- Windows Metafile (* wmf)
- DrawPerfect (* wpg)
- Micrografx Designer/Draw (* drw)
- Auto Cad 2-D (* dxf)
- HP Graphic Language (* hgl)
- Computer Graphics Metafile (*.cgm)
- Encapsulated Postscript (* eps)
- Tagged Image Format (* tif)
- PC Paintbrush (* pcx)
- Lotus 1 2-3 Graphics (* pic)
- AutoCad Plot File (* plo)

Koliko formata iz spuka će vam biti na raspolaganju zavisi od toga koliko ste konvertera instalira li. Ukolike ste izabrah punu instalaciju, na raspolaganju su vam svi formati. Kada se ne koristi puna instalacija, tada korisnik ima na raspolaganju mogućnost da sam bira potrebne formate iz spuka. Vi ćeta, naravno, odabrati one formate so kojima se srećete a svakodnevnom radu a mi ćemo spomenuti neka koje smatamo apsolutne potrebnima. To su drajveri za **WMF WPG, TIF** i **PCX** formate. Ova ko uvezena slika (kako rekosmo) po pravilu ne odgovara našim zahtevima – ili po dimenzijama ili po položaju, pa ju je potrebno naknadno podesiti Načine podešavanja možemo da podelimo na grube i precizna. Da bismo manipulisali slikom, potrebno je da je označimo To ćemo najlakše uraditi mišem, tako što ćemo kliknuti na nju, ali je moguća sliku označiti i uz pomoć tastature – postupak je isti kao kada se označava blok teksta. Tada se oko slika pojavljuje okvir, na kome se može primetiti šest značajnih tačaka. Kada je okvir vidljiv (znači, kada je slika selektovana), mogu se dalje vršiti operacije se njom.

Veličinu slike možemo **grubo** promeniti dovođenjem pointera na bilo koju od šest tačaka na okviru slika Tada se strelica menja u „dvostruku strelicu" i dalji rad je ista *Windows* logika – pritisak na taster, previačenje okvira na novu poziciju i otpuštanje tastera menja veličinu slike Kada se koristi ovaj način promene veličine slike, jasno je da se rad „od oka", tako se na statusnoj liniji sve vreme vidi odnos stare i nove veličine – mi jednostavno ne omogućava da vežbinu zadamo u jedinicama mere, što je često potrebno. **Sliku pomeramo** po dokumentu pomoću miša kao što pomeramo blok teksta – kliknemo i držimo taster na mišu, previačeno sliku na drugo mesto, i zatim otpustimo taster Mesta na koje će slika biti smeštena savisi od okolnog teksta – ukolike je umećemo u sredinu nekog reda tada savisi i od trenutnog preloma reda, a svaka dokucavanje teksta nemilosrdno manja položaj slike Sve u svemu – dosta brzo i lako ali i dosta grubo

Postoji i **precizniji način procene veličine** (i istovremeno još nekih parametara) slike Dijalog za podešavanja parametara slike se dobija pozivom **PICTURE** opcije **Fomat** menja Uz pomoć dijalo ga je moguće sliku povećavati ili smanjivati za određen procenat ili zadati dimenzije u jedinicama mere. Promenom jednog od parametara setati parametri dobijaju nove vrednosti (na primer, kada promenimo širinu slike, veličina u procentima se automatski izračunava i obratno). U ustom dijalogu imamo i mogućnost pomeranja slike u odnosu na margine, odnosno precizno zadavanje mesta slike na papiru Time eliminišemo neke nedostatke pomeranja mišem (pozicija je sada čvrsto određene), ali postoje i neka ograničenja. Naime, ukoliko želimo da na poziciju računa u odnosu na margine, slika mora da bude potpuno van teksta, može se pomeranje koje sma uneli odnosi na okolni tekst, a na na margine

POLJA

FIELD opcija **Insert** menija je veoma kompleksna. Ona umeće u tekst polje čiji se sadržaj automatski izračunava, pa se onde ubacuje u dokument. Sa nekim vrstama polja smo se već upoznali na pri-

mer, **numpages** je broj strana a **page** broj tekuće strane. Sada nam je lako da u svaki futer (dno strane) ubacimo polje oblika **"Strana" PAGE „od" NUMPAGES** (PAGE i NUMPAGES se, naravno, na kucaju nego biraju iz menija) i da imamo numerisanu stranu u obliku „Strana 5 od 12". Verovatno ste već shvatili kako program automatski numeriše stranu – u header ili futer ubaci polje za broj strane, tako formatirana će odgovarati našoj zahtevama. Kada gledamo spisak mogućnosti koje nudi polje, čini nam se da su autori *Worda* preterali u raznolikosti, ali ima i vrlo korisnih skrivenih mogućnosti. Kao primer moguće primene polja poslužite opcije za unose i kreiranje datuma. Ako pišemo neki standardni dokument (gde se ne fiksnota mesto nalazi datum), dovoljno je ubaciti datumsko polje – pri novom pisanju (ispravljanju) dokumenta, datum će automatski biti pravilno unet.

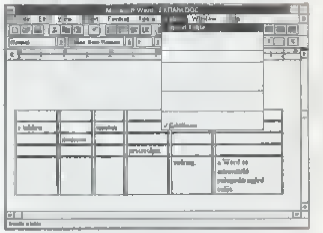
RAMOVI

FRAME opcija **INSERT** menija će napraviti prazan ram na ekranu ili će (što je još korisnije) uokviriti blok teksta koji smo prethodno označili. Ramovi su potencijalno najzanimljiviji, jer nam mogućnost pomeranja papiru kako nam odgovara i nisamo mišem ograničeni na raziku od reda bez ramova. Korisnici Venture su ovde na svojoj teritoriji – rad se ramovima im je nešto osnovno. Ipak, nemojte stavljati ramove tamo gde se može bez njih, jer time ne neki način naređujete računaru gde da smesti tekst, ne dozvoljavajući mu da slobodno slaže stranu. U ram je moguće umetnuti bilo koji element dokumenta – tekst, sliku ili objekat. U okviru **Format** menija takođe postoji **Frame** opcija. Uz njenu pomoć vršimo „fino podešavanje" rameva. Kao u svakom ozbiljnom tekst-procesoru, tekst može da „klizi" oko rama ili da se nastavlja tek posle njega. Ram je moguće postaviti relativno u odnosu na stranu ili pasus, a korisnike Venture će obradovati podatak da se svi parametri zadaju u jedinicama mera. Naravno, ramova je moguće premeštati uz pomoć miša, ali se profesionalce kojima je svaki milimetar pozicije važan, ovo je neurbešna opcija.

OKVIRI

Svakom elementu dokumenta je moguće dodati okvir. Tu mogućnost dobijamo posivom **BORDER** opcije **Format** menija. Ovo je veoma kompleksna ali iste tehike korisna opcija. Ona nam omogućava da obeleženom delu teksta (tablice, slike...) dodelimo okvir i šrafuru (raster) po želji. Ukoliko imate kolor štampač, sva šrafura može biti u boji. Nemojte okvire mešati se ramovima. Prvi pojam određuje **izgled** teksta, a drugi njegov položaj. **Border** najjednostavnije možemo da isprobamo tako što označimo blok teksta (sliku). Onda izaberemo okvir kakav želimo i tekst se nađe u njemu. Za svaki okvir moguće po želji birati ave elemente (na primer: ako uzmemo da su desna i donja ivica okvira deblja linija, a gornja i leva tanje, dobijamo okvir se „senkom"). Mogućnosti dizajniranja okvira su prepuštene vašoj mašti. Ako uokviravanje tablicu, već mnamo ozbiljnog posla oko planiranja. Za početak, zamislite da svako polje tablice može da ima svoje elemente okvira, različite od susednih polja. Sada, od vas zavisi koliko polja ćete obuhvatiti jednim blokom. Kada se tako obeleženi blok polja pozovete uokviravanje, dobijate isti dijalog kao za tekst okvir ali sada se dodatnim linijama po sredini polja, koje predstavljaju linije između obeleženih ćelija tablica. Kada ih ave podesite, ave obeležena polja odjednom dobijaju izgled okvire. Ovo možda zvuči komplikovano, ali nije – biće dovoljno da dve-tri tablice sredite i uokvirite po želji da biste dobili osećaj kako se to radi. Ka da želite da uokvirite polja tablice ili celu tablicu, trudite se da što više polja obuhvatite odjednom, a ne da radite polje po polje, jer ćete tada utrošiti dosta vremena. Pri kreiranju tablica najbolje ćete videti šte znači brzina računara – ništa ne usporava ovaj program kao kompleksna tablica se puno raznih okvira na ekranu.

...I RAD SA TABLICAMA



Lako i jednostavno: Rad sa tablicama

Da su tabele veoma važan dao obrade teksta vidi se i po tome što su autori *Word*-a njima posvetili ceo jedan meni. Po našem mišljenju: ovde se tabele slažu lakše nego u bilo kom drugom programu. Tabele ne moraju da imaju isti broj kolona u svim redovima, i obrnuto, a svaka ćelija može imati svoju veličinu, naravno, u meri u kojoj je to moguće uklopiti se ostalim ćelijama. Ćelije i grupe ćelija u tabeli se mogu odabrati na više načina.

- Jedna ćelija se bira tako što se miš dovede na levu ivicu ćelije koju želimo da označimo. Tada se strelica miša „okrenu" kao kada označavamo blok teksta zatim jednostavnim klikom miša ćelija biva označena
- Grupa ćelija se označava kao i jedna ćelija, osim što se sada taster na mišu (umesto jednog klika) drži pritisnut ave dok ne obuhvatimo ave željene ćelije.
- Red ćelija se označava kao i red teksta – dakle, pomerimo miša na samu levu ivicu ekrana, a pointer se okrena. Tada jednim klikom obeležimo čitav red. Ponavljanjem ave radnje, označavamo više redova.
- Kolona ćelija se bira tako što se pointer dovede na gornju ivicu tabele. Tada se on pretvori u strelicu koja pokazuje nadole. Jedan klik i – kolona je označena. Više kolona označavamo tako što „prevučemo" pointer (dok pokazuje nadole) preko željenih kolona.
- Cela tabela se označava tako što se pritisne Alt+broj 5 na numeričkoj tastaturi. Ali, ova se radi samo kada je NumLock uključen dok se i naša svi redovi ih kolone odabiraju posebno (se vano od oblika tabele). Označavanje delova tabele je veoma značajna operacija. Čak ni sadržaj menja nje fiksan nego zavisi od toga da li se kursor nalazi u tabeli, da li je odabran red tabele ih kolona.
- **INSERT TABLE** je opcija kojom kreirama novu tabelu. Broj redova i kolona biramo tako što ga unesemo u odgovarajuće polje (*column* je broj kolona, a *row* je red), a zatim i podstak o tome da li će širina redova u tabeli biti određena automatski (pri čemu se, u stvari, ave slobodan prostor podeli jednako svakom redu) ili će biti fiksna veličina.
- **INSERT COLUMNS** ih **INSERT ROWS** je opcija koja zavisi od toga koji element tabele je izabran. Moguće je umetati redove i kolone po želji, uz razne manipulacije kao što su šifovanje ostatka sadržaja u levo ih desno –
- **DELETE COLUMNS** ih **DELETE ROWS** je opcija koja briše kolona ih redova. Podržan je veliki broj manipulacija
- **MERGE CELLS** ih **SPLIT CELLS** će više obeleženih ćelija spojiti u jednu, ili podeliti ćelija koja je nastala spajanjem više ćelija. Može se desiti da vas objašnjenja poslednje tri opcije pomalo zbunjuju – ona su divojaka zbog toga što se tekst unesja u zavisnosti od toga u kom režimu reda se nalazite (unos teksta u ćelija, ili je selektovan neki deo teksta)
- **CONVERT TEXT TO TABLE** ih **CONVERT TABLE TO TEXT** će u zavisnosti od toga šta je obeleženo (tablica ili tekst) konvertovati i te tako što će tekst ubaciti u polja tablice, ili tablicu „pretvoriti" u običan tekst. Preporučujemo vam da izbegavate operaciju prelivanja teksta u tablicu, jer je ta konverzija uvek povezana se problemima (program greši u zaključivanju gde je kraj jednog polja i orda treba dosta truda da bi se sve postavilo na svoje mesto)
- **SELECT ROW SELECT COLUMN** i **SELECT TABLE** su opcije koje ćete pozivati samo likoliko nemate miša. Inače je mnogo lakše ova označavanje vršiti mišem.
- **ROW HEIGHT** je opcija kojom menjamo visinu reda u kome se nalazi kursor, ili, ako smo isa brali više ćelija, u celom bloku ćelija. Možete zadati tačnu ili **najmanju** visinu ćelija. Ukoliko ova opcija nije podešavana, sve ćelije imaju visinu koja se automatski izračunava, tako da sadržaj ćelije bude prikazan
- **COLUMN WIDTH** je opcija slična prethodnoj, se tora razlikuma što se sada određuje širina ćelija.
- **SPLIT TABLE** će preseći horizontalno tabelu na mestu na kome se kursor nalazi i napraviti dva. Ponekad korisno, ali se čuvajte slučajnog startovanja ove opcije. Ako ste greškom presekli tablicu, a zatim (opet greškom) niste odmah i skoristih „split table undo (Ctrt+Z)", imaćete muke da ave vratite u staro stanje
- **GRIDLINES** odrešuje da li će se videti kontura tablice čak i ukoliko ona nema okvir. Preporučujemo vam da ova opcija UVEK bude čekirana. Time eliminišete problem tipa „imam tablicu ali je ne vidim". Kako se tablica (ih nekan njenim čelovima) dodaje okvir, već sme pomenuh govoreći o format/border meniju.

FINALNI ZAHVATI

Do sada je trebalo da naučimo kakva pravila vešte sa slaganje jednog dokumenta.

- Pre samog pisanja iskoristite template koji vam je potrebna – posebno je poželjno da je veličina strana dobro definisana.
- Još dok unosite tekst, pošeljno je koristiti ave potrebne inpise (italic, bold...) jer je kasnije taše „pohvatati" ova mesta gde se pojavljuju reči koje želimo da istaknemo.
- Tek po završetku pisanja teksta (ake je to ikako moguće) koristite opcije umetanja elemenata i objekata u dokument. Naime, pri radu se dokumentom koji barata većim brojem slika i formula dolazi do usporenja. Naknadnim ubacivanjem elemenata se usporenje eliminiše.
- Pri samom finalnom slaganju dokumente šte vaše konsultujte **Preview** opciju, a kojoj ćemo više reči u sledećem poglavlju

TAMPANJE

Štampanje je obično krajnji cilj pisanja teksta. Da bi štampanje bilo uspešno izvedeno potrebne je složiti dokument (što smo već naučili kako se radi) i podesiti štampač (što ćemo tek videti). Osim prostog slanja dokumenta na štampač, postoje i kompleksni ja operacija spajanja podataka pa zatim štampanje – **print merge**.

PRINT MERGE omogućava da u jednom fajlu definišemo neka podatke (imena, adrese) a u drugom standardni dokument na koji ćemo „lepiti" te

podatke. Pošto ovo zvuči komplikovano, objasnićemo na primeru. ručno da morate neko poslovne pisamo da pošaljete na dvesta adresa. Pisma je standardno a jedina stvar koja se menja su adresa. Da ne bismo svaki put kucali isti tekst, jednom zauvek definisati pisma i u okviru njega mesto na koje će se ubacivati adresa. Onda ćemo pomoću **print merge** opcije povezati ovaj tekst sa bazom adresa, a za ostale će se pobrinuti program. Ukoliko unate te adrese u nekom drugom formatu podataka (*dBase?*) morate ih prethodno konvertovati u *Word 2.0* format. Ove opcije je dosta praktična, ali se u našim sloviima retko koristi.

PODEŠAVANJE ŠTAMPAČA

Štampač iz *Worda* podešavamo tako što pozivamo **Print Setup** opciju iz **File** menija. Tada dobijemo dijalog za podešavanje štampača koji je deo samih *Windowsa*. Kakav će se dijalog za podešavanje štampača pojaviti, zavisi od samog drajvera za štampač. Uglavnom, kod svih štampača je moguće podesiti veličinu strane, margine i slično. Potrudite se da vam **uvek** budu akladno podešeni ugled strane (**Page Setup**) i štampač (**Print Setup**). Time ćete izbeći situaciju da „dokument savršene izgleda na ekranu, ali nikako ne može da se lepo odštampa.

PREVIEW DOKUMENTA

PRINT PREVIEW je opcija kojom su se tekst editori u tekst-modu služili da bi mogli dočarali kako će izgledati odštampan dokument. Iako nam *Word* u svakom trenutku prikazuje kako će izgledati deo strane na kojoj trenutno radimo, korisno je pogledati i celu stranu. Kada pozovemo preview i dobijemo pregled cele strane, imamo nekoliko mogućnosti. Možemo odštampati tekst ukoliko smo njime zadovoljni (bez izlaska iz previewa i izlaske u print opciju). Možemo pogledati i podesiti margine, i to na veoma jednostavan način. Možemo uključiti istovremen pregled dve strane ili jednostavno izaći iz preview opcije. Po tekstu se kretamo pomoću skrol bara na desnoj strani ekrana ili pomoću **PageUP/PageDN** kombinacije tastera. Ako nismo zadovoljni marginama moguće je povrati **Margins** opciju, pa zatim mi šem podesiti margine.

PRINT...

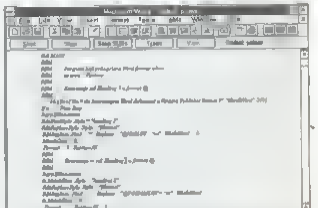
Pozivom **Print** opcije **File** menija (ili pritiskom na **Ctrl+Shift+F12**) ulazimo u dijalog za štampanje dokumenta. Dokument je moguće poslati delimično ili ceo, u više kopija, na štampač ili u fajl. Takođe je moguće i iz ovog dijaloga pozvati podešavanje štampača, što je neka vrsta poslednje šanse – ako primetite da niste podesili štampač tek kada je štampanje počelo zakasnili ste. Štampanje u fajl je dosta korisna opcija – odštampate ceo dokument u fajl, zatim taj fajl odnesete na drugi računar sa kvalitetnim štampačem i obavite besprekorno štampanje. Kada radite ovakve operacije morate biti sigurni da sta dobro instalirali i lepo podesili drajver za štampač.

MAKRO JEZIK

Već smo spomenuli mogućnost kreiranje makro naredbi. Makro naredbe su pravi mali (ili malo veći) programi koji obavljaju neke ustaljene opcije. U slučaju tekst-procesora te ustaljene operacije su, recimo konvertovanje teksta iz jednog u drugi kodni raspored ili standardno formatiranje teksta kada ga uvezemo iz nekog drugog formata. Naravno, sve te operacije je moguće uraditi i „ručno", ali ako neki postupak ponavljate, razumno je jednom se pomučiti i napisati program koji će sam obavljati tu operaciju. Ukoliko se postupak koji želimo da automatizujemo može izvesti preko opcija iz menija, ne moramo ni da pribegavamo programiranju. Dovoljno je da naredimo računaru da zapamti operacije koje obavljamo, a da ih zatim sve ponovi kada nam je to potrebno.

Sada ćemo da kreiramo jednostavnu makro na redbu:

- ☐ Pozovite **Record Macro** opciju iz **Tools** menija.
- ☐ U dijalog-boks unesite ime makro naredbe, kao i kombinaciju tastera kojima želite da je te naredbe startuje.
- ☐ Uradite sve operacije koje želite – na primer, pe-



Automatizacija zamornih poslova. Makro jezik

zovite opciju **Symbol** iz **Insert** menija, a zatim izaberite neki karakter koja je nemoguće dobiti sa tastature (ovo pomoću kombinacije Alt+slovo)

- ☐ Kada ste sve željene operacije izvršili (a te može da bude čitava serija, znatno komplikovanije od našeg primera), pozovite **Stop Recorder** opciju **Tools** menija. Time ste nalošlih programu da prestane sa zapamćenjem vaših poteza i da je makro naredba izvršena.

Pošto smo formirali makro komandu, želećemo da je izvršimo i da vidimo kako radi. Sada ćemo ignorisati kombinaciju testera koje sta definisali (ako sta ih definisali) kao „prečicu" za start makro naredbe i pozabaviti se opcijom iz menija. **Makro** opcija **Tools** menija nam omogućava razne usluge – start neke od makro naredbi ili izmenu i debagovanje istih. Kada iz dijalog-boksa izaberemo određenu makro naredbu, opcije na desnoj strani dijaloga postaju aktivne (osim se izabrali). Osim dosta nevažnih operacija se samim programima (kao što su brisanje i preimenovanje) tu su i dve veoma važne – opcije **Run** i opcija **Edit. Run** nam služi da iz spiska makro naredbi odaberemo jednu i startujemo je. **Edit** opcija je toliko bitna da ćemo je obraditi detaljnije. Ukoliko na **Edit** (da li ste izabrali ime makro naredba iz spiska?) ulazite u editor makro naredbi. Na primer, izabrate maločas kreiran makro koji bira slova iz **insert/symbol** menija i kliknite na odbt. Makro naredba je, kako vidimo, tekstualni program – zapravo, svaku naša makro naredba kreirana napisane opisanim postupkom sa prevedi u tekstualni oblik. Program može izgledati ovako

```
Sub MAIN
InsertSymbol Font = „Symbol", CharNum =
„229"
End Sub
```

Ako malo bolje pogledate „program" koji je kreiran, videćete da dosta liči na *Basic*. Time smo dobili do drugog načina kreiranja makro-naredbi – *WordBasicu Word for Windows* je i u prethodnoj verziji imao mogućnost formiranja makro naredbi, ali je tek u sadašnjoj verziji pisanje programa dovedeno do savršenstva. Svi već tekst-procesori takođe imaju mogućnost rada sa makro naredbama, ali je uvek makro jezik dosta komplikovan i specifičan. Po prvi put kod *Words for Windows* sa makro naredbe pišu u običnom bejziku!! Time se ovaj tekst-procesor izako izdigao iznad konkurencije da u prvi mah čovek i nije svestan svih njegovih prednosti. Kao prvo, zamislite prosečnog korisnika tekst-procesora, ako je ekspert u oblasti slaganja teksta, običan operater ili pisac, on ne mora da zna (i nauverovatnije ne zna) da programira. A opet, retki su oni koji su se sretali sa računarima a da ne znaju bejzik. Time je omogućeno širokom krugu korisnika da sa svoje potrebe napravi makro naredba kakve su im potrebne. Znamo kolike su „Microsoft" kao firma i posebno njen predsednik Bil Gejts vezani za bejzik. Zato smo sigurni da su uradili dobar posao napravivši makro jezik koji je u stvari *basic-like*. Sam program se piše na najstandardnaji mogući način. Imaje mogu ali ne moraju da imaju brojeve, a labels se koriste sa kontrolu grananja programa. Ukoliko se do sada niste sretali sa labelama, dećemo kratko objasniti: labela je reč od nekoliko slova koja se završava dvotačkom (na primer **labela:**), a kasnije se skak na labelu vrši naredbom **Goto labela**. Program može i

imati procedura i funkcija, parametri se prenose na standardan način, a posebna prednost je objektna kontrola *Windows* elemenata – ovaj bejzik veoma nalik na *Visual Basic* iste firme. Promenljive ne moraju da se deklarišu, nego se, kada vam to zatreba, jednostavno upotrebe. Postoje čak i **print** i **input** naredbe, pa se možete ovim bejzikom po želji igrati pogađanja brojeva ili slagati tekst.

OSNOVNE STVARI...

Sigurni smo da vam je već poznato kako se pišu programi u bejziku, tako da se nećemo previše zadržavati na tome. Neke naredbe postoje u bilo kom bejziku – na primer **For...Next** petlja, **IF** i **GOTO** naredba i još mnoge druge. Sintaksa i namena ovih naredbi je toliko standardizovana da je dovoljno da poznajete bilo koji bejzik (počev od Spektrumovog) da biste radili sa *WordBasicom*. Izuzetak je naredbe koje se odnose na sam rad pod *Windows* okruženjem. One nisu standardni deo bejzika, ali se vam biti poznate ukoliko ste radili sa *Microsoft Visual Basic*-om. Tako na primer, **MsgBox** daje standardni *Windows* upitnik (tasa „yes, no, cancel"), **Shell** izvršava drugi program, i slične. Ove funkcije se laka koriste, a kako je jedna jednostavne možete videti u primeru koji smo dali. Ovaj bejzik se pridržava objektnih pravila koja je uveo *Visual Basic*. Tako mi šete primeniti radnje na koji se primjene varijabilne statemash naredbi. Na primer **Search**ili**pisanje Find = "xxx"** skriva «vašav pesiv promenljive. Kada jednom savladate bejzik i pomalo radine objektnog razmišljanja, rad sa *WordBasic*-om postaje lak i zabaven.

...I SVE OSTALO

Na žalost, „sve ostalo" je mnoge komplikovanije i obimnija oblast. Pod ostalim naredbama podrazumevamo sve naredbe koje se odnose na uređivanje teksta. Pošljene su po oblastima, a pri pisanju programa osta verovatne morati da dosta konsultujete indeks. Pošto postaji veliki broj naredbi, se obimnim brojem parametara, izbegavajte „ručno" unošenje i koristite snimanje makroa. O ovome će biti više reči malo kasnije, a sada da vidimo kako ćemo se snaći u indeksu. Prvo, moramo znati koju operaciju se tek stora želimo da izvedemo. Zatim, u indeksu pronađemo grupu naredbi koja nas zanima (na primer, „naredbe i funkcije za rad sa blokovima"). U okviru grupe pronađemo naredbu koja nam je potrebna i na kraju eventualno pozovemo help radi sintakse. Ova procedura je mnogo brža nego što izgleda, zahvaljujući pomoći koju nam nudi *Word*. Naime, ukoliko se nalazite u modu izmene makro naredbe (u Edit Macro prozoru), pritisak na **F1** vam daje pomoć za naredbu na kojoj je kursor. Ta pomoć je vrlo korisna i (za *Windows* okruženje) brza. Ukoliko pri pisanju makro naredbe ne uklanjate *Help program* iz memorije nego ga smanjite na ikonu dobijate pomoć u najkraćem mogućem roku.

KAKO PISATI PROGRAME?

Programe možete pisati na tri načina. Snimanjem akcija (**record macro**), pisanjem programa ili kombinovanjem prethodna dva načina. Mi vam apsolutno preporučujemo kombinovanje, i to iz više razloga. Na prvom mestu, sasta je malo stvari koje se ne mogu izvesti putem menija. Zatim, snimanje akcija nas osigurava od logičkih ili slovnih grešaka u programu. Na kraju je korisno dodati kontrolne strukture, tako da se program izvršava onoliko puta koliko nam je potrebno. Da bismo ilustrovali pred nosti kombinovanog pisanja, iskoristićemo primer „iz života": bilo je potrebno konvertovati veću količinu teksta iz *XY Write* formata (tekst format) u *Word* format, odnosno, trebalo je konvertovati sve reči koje je *XY* ne svoj način markirao kao *bold* u normalni *bold* (takođe i italic, underlined.) upis u *Wordu*. Kako je organizovan *XY* dokument? Podebljan tekst počinje sa "*<<MDBO>>*", a završava se sa "*<<MDNM>>*" Znajući te, uradik smo sledeće:

- Prvo smo startovali snimanje makroa (macro recording)
- Pozvali smo **Find** opciju **Edit** menija, a iz nje startovali traženje *"dMDNMdZ"* stringa.
- Kod prvog nađenog stringa smo izveli seriju operacija (sa tastature) kojom smo „očistili" format stringove (koji *Wordu* ne trebaju), a ostatak teksta ignorisali pri pretvorbi u bold.
- Prekinuli snimanje makro naredbi.
- Dodali **For..Next** petlju, kojom smo zahtevali ponavljanje čitave operacije 1000 puta. Šta se sa da desiava? Pošto u tekstu sigurno nema 1000 *bold* reči, dokument će do kraja biti prečišćen, a onda će nas *Word* pitati da počne pretraživanja od početka. Jednostavno ga prekinemo pritiskom na **Esc** i to je kraj rada makroa.

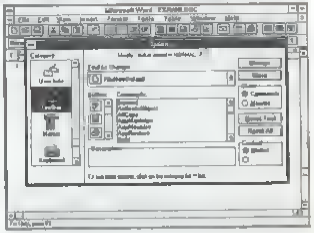
Iz ovog kratkog primera smo mogli da vidimo više stvari Prvo to da je program, iako jednostavan po svojim akcijama, komplikovan za pamćenje pa ga je (bar jedan deo) bilo lakše snimiti kao akcije Druga je stvar sasiuga logika – na morate pamtiti (posebno kao makro naredbi koje se za internu upotrebu) da li je neka akcija strogo precizno postavljena. Ono što želimo da kažemo je sledeće: iako smo mogli da detektujemo kraj teksta i prekinemo rad makroa na „legalan" način, to nije bilo neophodno, jer ga je lako prekinuti po potrebi, u interaktivnom radu.

```
Sub MAIN
For g = 1 To 1000
EditFind .Find = "<<MDBO>>", .WholeWord = 0, .Match
Case = 0, .Direction = 1, .Format = 0
EditClear
ExtendSelection
ExtendSelection "<<"
CharRight 1, 1
CharLeft 2, 1
Bold 1
CharRight 1
CharRight m, 1
EditClear
Next g
```

PODEŠAVANJE WORDA

Pošto smo u poglavlju „kako početi" obradili korak po korak kako podešavanje osnovnih parametra rada, sada ćemo dati pregled i ostalih podešavanja. Pri tome pretpostavljamo da ste već naučili neke operacije sa *Wordom* i *Windows*-om, pa nećemo previše ulaziti u detalje U podešavanje *Worda* se ulazi tako što se u meniju **Tools** dobere opcija **Optinas (Alt+O,O)** Dobija se kompleksan dijalog-boks, ali je važno shvatiti osnovni princip rada na levoj strani se nalazi vertikalna linija sa ikonama, koja skroluje gore-dole, a četvrtak dijaloga se menja u zavisnosti od toga koju ikonu odnosno koje podešavanje vršite Krenućemo redom:

- **VIEW** određuje pojavljivanje određenih elemenata slika, teksta i okvira. Malimo da su sve opcije dobro podešene i da nema potrebe na njihovom menjanjem
- **GENERAL** nam omogućava da podesimo opcije „od opšteg značaja", npr jedinice mere ili repaginaciju strana u pozadini Jedinica mera smo podesili još na početku, a ako uključimo **Background Repagination** eliminisaćemo potrebu **Repaginate Now** opcijom, jer su prelomi strana uvek svaki izračunati
- **PRINT** određuje način štampe (draft ili kvalitetan), dodavanje informacija u dokument i slično..



Prilagoditi program svojim potrebama: Podešavanje Worda

- **SAVE** određuje parametre štampe. **Allways Create Backup Copy** bi trebalo da bude uključeno, kao i opcija **Automatic Save Every** uz koju program svakih n minuta (kod nas 10) snimi *backup* kopiju teksta. U slučaju nestanka struje tada gubite samo poslednih nekoliko minuta rada.
- **SPELLING** podešava opcije sa proveru „spelovanja"
- **GRAMMAR** vam nudi izbor između tri stepena provere gramatike
- **USER INFO** vam omogućava da upravite ili unesete svoje podatke, ukoliko to niste učinili pri instalaciji
- **TOOLBAR** je već zasvim ozbiljna opcija – ona nam omogućava da potpuno prilagodimo *toolbar* našim potrebama. Svaku ikonu je moguće premestiti postaviti na drugo mesto i za nju zakačiti makro program ili funkciju po želji Redefinisanje *toolbar*-a se tako nauči, pa nema potrebe da se detaljna objašnjava. Uostalom kada dođe do toga da menjate *toolbar* već ćete i sami znati kako se to radi. Jednostavno, odaberete mesto na *toolbar* linji, ikonu koju stavljate na to mesto, program ili naredbu koja će se izvršiti klikom na to mesto i to je sve.
- **MENUS** je takođe kompleksna opcija koja nam omogućava da u menije *Word* e ubacimo programe i naredbe po želji. Ovde je postupak za nijan su složenji nego kod *toolbar* linije, ali je i dalje dosta jednostavan izaberete mesto u nekom od menija koje definišete zatim izaberete makro program ili naredbu i meni je promenjen. Druga korisne primena ove opcije je brisanje opcija iz menija – mi smo uklonili pomoć za *WordPerfect*, koja je u našem slučaju suvišna. Ova opcija je (kao i prethodna) pravi način za instaliranje vaših makro programa. Kada često koristite naku makro procaduru, start iz menija (ili sa *toolbar* linije, što može biti još bolje) je najbrži mogući način i štedi dosta vremena.
- **KEYBOARD** je opcija kojom redefinišemo tastaturu. Nije od preveliks koristi, jer je ogram čen izbor tastera koji se mogu redefinisati. Ipak, ukoliko želite neki makro da startujete preko tastature (mada se to može podesiti i u **Macro** opciji **Tools** menije) ova opcija će to omogućiti
- **WIN INI** omogućava da menjame čitave istu menog fajla Ovu opciju nemojte koristiti, ukoliko tačno ne znate šta radite! Sve promene koje napravite postaju stalne – zato, opres se preterana nim eksperimentisanjem.

PAR SAVETA ZA KRAJ

Najrednijim korisnicima verovatno nisu potrebni saveti, ali ih podsećamo na nekakvo važnih stvari koje im mogu pomoći da poboljšaju rad. Prva stvar je zauzeće mesta na disku – iako je primamljivo instalirati ceo program, ja može da se svede na pet megabajta bez preteranog gubljenja na mogućnostima. Pre svega, obrišite sve nepotrebne **.dot** fajlove – svi templejti, osim **normal.dot**, su suvišni. Drugo, dobro razmislite da li vam je potrebna kontrola spelovanja i rečnik sinonima – ukoliko se pišete na zaglesikom ovi fajlovi samo zauzimaju mesto na disku. I na kraju, eliminišete dupliranje funkcija – ukoliko imate (i koristite) *CorelDraw* ne treba vam *DrawEditor*, a ukoliko imate *Excel* ne treba vam *Microsoft Graph* –Obavemo uključite automatsko snimanje i to na nekoliko minuta. Nama se desilo da se računar resetovao kad u toku pisanja ovog uputka (a tako nam i treba, kada startujmo nesigurne DOS programe u *Windows* prozoru, paralelno sa pisanjem teksta) Nakon ponovnog startovanja, *Word* je raznašljao i rezmuljao, pa na kraju učitao dokument u čijoj je naslovnoj linji pisalo „Microsoft Word-WINWORD DOC (Recovered)" Ovo *recovered* smo videli prvi (i nadamo se poslednji) put, ali ono što je spasilo (gotovo) ceo tekst je bilo često snimanje I, na kraju, pri radu koristite **WKEY** drajver za tastaturu na *Windows*, o kome smo pisali u „Računarima" u rubrici o *Windows* programiranju Te je jedno i najbolji *Windows* drajver za tastaturu Može se naći i na Sezamu. Omogućava definisanje tastera po želji i u bilo kom kodnom rasporedu Ovaj program je po mogućnostima prava suprotnost hakerisanju po **.dll** bibliotekama.

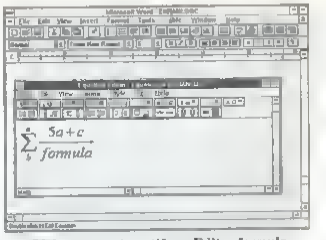
DODATNI PROGRAMI

Ako ste instalirali kompletnu verziju *Worda*, onda vam se na disku nalaze i četiri programe koji se posebno instaliraju Ovi programi su radem tako da posiuju filezofiju objektnog linkovanja podataka. Nevost koju smo ovde prvi put videli je način na koji ovi programi rade iako su to kompletne *Windows* aplikacije, nemaguća ih je startovati odvojeno, kao posebne programe, nego samo kao servere za objek te Te znači da svaki program (e ne samo *Word*) koji ima kao opciju uvoz objekata, može pozvati neki od ovih programa, obraditi podatke u njemu i uvesti ih u sebe Ovim potezom je „Microsoft" još jednom dokazao da je lider, ako ne na DOS polju (gde puno nezavisnih proizvođača ima bolja programska rešenja od njega), onda sigurno u klasi *Windows* programa. I pe idejama i po izvođenju, ovi programi su nešto nove. Druga važna činjenica je posao kap ovi programi rada – se male nepera, eni mogu da zamene vile programa drugih firmi (*CorelDraw*, *Microsoft Excel*) koji dosta koštaju Dovoljno je malo praćkanati po videti da sena *Words for Windows 2.0*, ma koliko nam visoka izgledala, nije ni blizu zbiru cena nekog jeftinijeg tekst-procesora (sa manje mogućnosti) i dodatnih programa, a da i ne govorimo koliko u radu sa *Wordom* uma vile elegancije Prvi program (editor formula) ćemo detaljno prikazati, jer na po-

stoji program koji bi ga zamenio – formule se služu samo editorom formula i nijednim drugim programom. Ostala tri programa služe kao zamena za neke ostvarujuće programe, pa ćemo ih malo slobodnije prikazati ..

EDITOR FORMULA

Editor formula možete startovati na dva načina. Ili tako što ćete iz menija *Insert/Object/Equation* startovati server aplikaciju za objekat „formulu", ili tako što ćete (ukoliko ste je instalirali kako smo vam preporučili) izabrati ikonu sa *toolbar* linije. Pošto je ovaj program, kako smo već rekli, ponešto lasa klasične *Windows* aplikacije, on ima i sve elemente *Windows* a. Meni nam omogućava da izaberemo stilove kojima se ispisuje formula i iz menija često u glavnom radi razna podešavanja. Glavni rad se odvije sasvim interaktivno pomoću miša i „tastera" koji se nalaze ispod meni-linije. Moramo odmah da kažemo da bolje urađen sistem za slaganje formulisnome do sada videli. Način na koji se formula slaži u da sada take hvalevnim programima kao što su *Ventura* ili *WordPerfect* nisu ni blizu lakoći i mogućnosti kakvu u radu nudi *Equation Editor*. Pozne te je kako su do sada radili paketi za slaganje formula (stalno poređimo ovaj deo *Worda* sa opcija ma iz *ventura* i *WordPerfecta*, jer su ova dva programa do sada nudili najviše u svoj oblasti). Problem je bio u tome što se podrazumevalo poznavanje matematike ih bar nazive svih elemente koji su potrebni da bi se sastavila formula (zagrade, razlomci, korenovanje i stepenovanje.) Drugi korak je bilo učenje slaganje formula – vi biste umeili elemente formule (i to tekstualno!) dok bi program pokušavao da zaključi kakvu vi formulu u stvari želite. Taj posao je bio toliko mučan, da su formule u venturi sastavna vrhuncem slagačkog umeća. „Microsoft" je i do sada imao praksu da od drugih firmi naručuje delove svojih programa, ukoliko su rešenja koja te firme nude dovoljno dobra. Firma „Design Science" je očigledno ispunila standarde „dobrog programa" pa i više od toga – smatramo da je jedan od najboljih potezaa „Microsoft"-a to što je u *Word* ugrađen baš njihov program za slaganje formula. Pristup samom slaganju je toliko različit od dosadašnjih rešenja, da ni smo verovali kako je sve to u stvari jednostavno kada smo počinjali da radimo sa ovim paketom. Ali, zar nau sva genijalna rešenja jednostavna? Veliku pomoć pri slaganju su obeležili vektorski fontovi Ventura je tekst slagala take što je koliko je to moguće, koristila prigodnje karaktere za delove formule. A zamislimo samo jednu situaciju unate jednostavnu formulu – samo jedan koren i vrednost ispod njega. U zavisnosti od toga šta je ispod korena, en će imati određenu visinu. Ukoliko ispod korena unacimo neki razlomak, visina korena se povećava dvostruko (ili više). E sad, ukoliko nemate fontove u svim veličinama potrebnim venturi (a ona, sećate se, koristi bit-mapirane fontove) na velikim stemukaran. Za razliku od toga, *Equation Editor* će zatražiti željenu veličinu fonta, *Windows* će ga generi-



Idelmno za matematičare Editor formula

sati i svi elementi formule će se završeno uklapati. Na trenutak ćemo ostaviti sam posao slaganja formula po ekranu i pogledati šta se nalazi u meniju editora formula.

- **FILE** meni ne sadrži, kako mu ime kaže, opera cije sa diskom, nego služi za uvoz složene formu le u *Word* i izlaz iz programa.
- **EDIT** meni ima standardne opcije za **cut/paste** i obeležavanje formula. Formulu je moguće obeležiti i mišem (i te lakše), pa ćete **select all** koristiti ako nemate miša.
- **VIEW** nam omogućava da biramo u kojem uvećanju ćemo gledati formulu. U većini slučajeva, formule nasu praviše velike, pa je pogodno gleda ti ih uvećane. Ukoliko imate neku ogromnu formulu možete je smestiti da „stane no ekran" ili posmetrati deo po deo. Ova povećanje nemaju veze sa stvarnim izgledom formula u dokumentu i služe samo da bi olakšala rad u toku samog slaganja.
- **FORMAT** je opcija koje služi raznim vrstama u ravnavanja matrica ili „gomilica" (*pile* u terminologiji *Equation Editora*) Naravno, uravnavaju se horizontalno i vertikalno, sa određenim prorednom.
- **STYLE** je najkomplikovanje opcija u celom meniju. Stilovi ni u *Equation Editoru* nisu izgubili svoju funkciju koju su imali u *Wordu* Ipak, stilovi su razdvojeni, to jest ovde važa sasvim drug stilovi i sasvim drugačiju načinu definisanja. Postoji sedam stiluva, a mnki treba koristiti u određenim prilikama.
- **Math** stil se dodeljuje ih izabranom delu formu le ih karakterima koje trenutna kucate Kada kreirate formula, koristite Math stil kao osnovni, a ostale stilove samo kada je to potrebna O vaj stil pravi razliku između funkcije i varijabli. Kako ovaj stil formatira tekst možete videti na primeru unesite e=ce (5) i videćete da je x upi san ukošenim slovima, a na (pošto je program prepoznao da je reč o formuli) normalnim
- **Text** stil ćete koristiti kada želite da kucate ab-

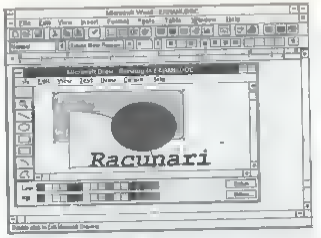
čan tekst, bez mešanja programa. Inače će program delove koji mu „liče" na funkcije ustvari, što kada se kuca običan tekst nije poželjno
- **Function.** Ovaj stil ćete dodeljivati izabranom delu formule samo u slučaju da je program „prepoznao" u zaključavanju, odnosno ukoliko ne prepozna funkciju
- **Variable** je stil koji ima ulogu suprotnu prethodnom Ukoliko ste unali varijablu koju je program prepoznao kao funkciju, dodeličete joj stil variable
- **Greek.** Postoje dva **Greek** stila- jedan za velika i jedan za male grčka slova. Koji stil će biti aktivan zavisi, naravno, od toga koje slovo je upotrebljeno **L.C. Greek** stil se pridružuje malim grčkim slovima, a **U.C–Greek** velikim Ako želite da otkucate samo jedno grčko slovo pritisnu ćete Ctrl+G, a zatim i to slovo, a program će tom slovu dodeliti **Greek** stil i opet uključiti prethodni Ukoliko želite da škarate **Greek** stil, pritisnite **Shift+Ctrl+G** i nadalje ćete raditi u **Greek** stilu
- **Matrix Vector** dodeljuje matrični/vektorski stil izabranom delu formula ih karakterima koje kucate **Ctrl+B** sledećem otkucanom slovu dodeljuje ovaj stil. a ako pritisnete **Shift+Ctrl+B** ovaj stil je fikuran
- **Other** je žokar stil, stil koga ćete koristiti po želji za neke specijalne namene
- **DEFINE** nam služi da predefinišemo stilove koje smo nabrojali Kada startujemo ovu opciju, dobijamo dijalog u kome biramo fontova i njihov način ispisa (iskošene ih podebljana slova) Sa mom instalacijom *Word-a*, ovi stilovi imaju dobre izabrane vrednost. Ipak, ukoliko ste (prema delu uputstva koje se odnosi na instalaciju pre samog rada) ukinuli postojanje *Times New Roman* fonta, biće potrebne da sami odredite font umesta njega. Kada je *Times New Roman* ukinut, program pokušava da ga sam zameni odgovarajućim fontom, ali je to kod nas resultovalo uključivanjem *Tomes* ćirilice.
- **SIZE** opcija određuje veličine delova formula Da pojasnimo svaki deo formule (indeks eksponenti, normalan tekst.) sa ispisuje u određenoj veličini Ta veličina je standardne dosta dobro odabrana, ali je ostavljena mogućnost da neki parametar promenite ukoliko vam ne odgovara. Veličina se, naravno, zadaje u tačkama i unosi u određeni dijalog-boks.
- **HELP** ustam ćemo samo pomenuti kao čobar i pregledan (ali nepotreban, čim steknete osećaj za slaganje formula) Formula se slaže take što se interaktivno dodaju elementi formule Pri tome nije potrebno da znata kako se koji element nove, dovoljno je da znate kako formula treba da izgleda Recimo da hoćete da složite neku jednostavnu formulu. Pošto pozovete *Equation Editor* naći ćete se u njegovom okruženju Sve što kucate će se pojavljivati na radnom delu površine. Ako otkucate „x=" te će se, naravno, pojavi-

ti. Sada želimo da ovu formuli dodamo desnu stranu Ispod glavnog menija se nalazi linija sa ikonama. Ne svakoj ikoni se nalazi dosta prepoznatljiva slika simbola koji se nalaze u okviru te ikone. Možete odmah prepoznati ikonu za zagrade ikonu za razlomke i koren ili ikonu za grčka slova. Tim načinom ćete lako „pogoditi" kakve simbole krije ostale ikone. Sada kliknu na ikonu kvadratnog korena bramo simbole koji su nam potrebni Recimo da smo izabrali ikonu koja una dva polja za unos kojom su predstavlja, na primer, „treći koren iz dva" Sa pomoću polja za u nos se sređeno prvi put. Polje za unos (ili *slot* u terminologiji *equation editora*) je osnovni elemnat širenja formule. Na početku unosa formuli imamo jedno polje za unos, u koje možemo uneti više stavki, ali samo horizontalno poredanih (neoa nikakvog grananja). Ako sada unesemo neki simbol, on može imati jedno ili više polja za u nos. Slot (ili slotovi) se mogu nalaziti na različi tim mestima u odnosu na simbol – kada izaberemo zagrade, slot se nalazi između leve i desna zagrade, a kada izaberemo razlomak, ima mo dva slota, iznad i ispod razlomačke crte. Lepote rada sa ovim editorom formula je to što je polja za unos moguće neograničeno popunjavati novim simbolima, a i ti simboli mogu imati svoja polja za unos. Time je teoretski moguće slagati neograničeno velika formula Pri dodavanju navih elemunata u formulu eni delovi koji (uslovno rečeno) ugkuviju doo formula se šire po potrebi Ukoliko imamo kvadratni koren i u okviru njega unosimo razlomke i komplikovane i sraze u zagradama, taj koren će se povećavati, tako da će uvek da obuhvata ceo izraz. Malo oprezności pri unosu je ipak potrebno Na primer, ukoliko primetimo da je neki deo formule treba lo da m nade u eksponencijalnom delu izraza e nije, moraćemo da obrišemo deo formule i dodamo polje za eksponent, pa tek onda nastavimo rad. Deo formule je moguće obeležiti jednostavnim „razvlačenjem" pointera dok je pritisnut ta ster na mišu. Ako pritisnemo **Delete** taster dok je neki deo obeležen, en će jednostavno biti obrisan.

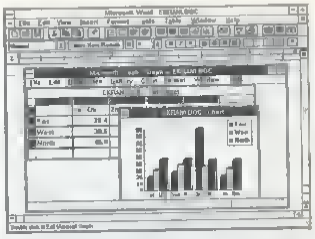
Time smo obradili osnovne operacija potrebne za kreiranje formula. Kako je već slučaj kod grafičkih mućnih programa, neke operacije je lakše naučiti nego ih objasniti, jer kada budete složili samo jedau formulu, umećete da složite sve ostale, ma kako komplikovane bile

EDITOR CRTEŽA

Editor crteža je program koji bi potencijalno mogao da zameni vaše *Draw* programe kao na primer *CorelDraw*: naravno, u slučaju da nemate prevelike zahteve Što se nas tiče, ovaj program je sasvim dovoljan. Program se startuje preko **Insert/Object/Microsoft Drawing** menija (ili kliknom na ikonu iz *toolbar* a. Stertom programa, otvara se prosor sa nekoliko površina. Na levoj strani se nala-



Editor crteža



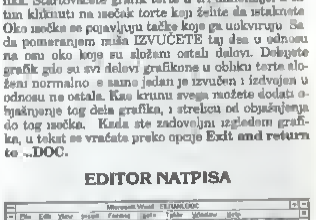
Editor grafika

u površina se slatina – ne postoji previše alata ali i ovaj izbor je solidno urađen – razni oblici krivih, površine i tekst. Dole se nalazi površine sa izbor boje objekata, gore meni, a u sredini radna površina – ukratko, samvim standardan izgled za ovu grupu programa. Moguće je uvoz slika iz drugi programa i dalji rad se njima. Podržane su operacije se objektima, već standardne u Draw programima, za redanje objekata jedinog preko drugog. Što je veoma važno se krajnji izgled crteža. Operacije se tekstom se takođe standardne – tekst se smatra delom crteža, tako da se i on uklapa u koncepciju rada se ovim programom.Ukratko, sve opcije su veoma standardne i logične, pa čete ih koristiti čak i ako znate da radite samo sa nekim *Paint* programom – na primer se *Paintbrush*-om koji standardno ide uz *Windows* –

EDITOR GRAFIKA

Microsoft Graph je program koji liči na svog velikog brata – *Microsoft Excel*. Ali samo liči. U programa su uključena samo neophodne opcije, dok se ostale služi sam *Excel*. *Graph* se na može svrstati u programe za tabelarne proračune, iako tako izgleda – on služi samo za prikazivanje tablica i tag deo posla obavlja veoma dobro. Kada se startuje ovaj program (preko opcije **Insert/Object/Microsoft Graph** ili klikom na *toolbar*), dobijamo jednostavnu organizaciju. ekranom dominiraju dva prozora – jedan služi za menjanje tabele (**Datasheet**), a drugi za prikaz grafika (**Chart**) Podatke možete unesti i uvezom iz *Excel*-a, a ne samo kucajući ih u tablicu. Program prikazuje podatke organizovane u dvodimenzionalnu tablicu, tako da uvek imamo dve ose i dva podale podataka. Uvek se jedna osa koristi kao glavna a druga kao sporedna – koja je koj određujete korišćenjem Data Series menija. Grafik možete prikazivati na puno načina, u trodimenzionalnom i dvodimenzionalnom prikazu. Svaki element grafika (svaki deo prikaze) može biti specifično obojen ili uokviren a sve se podešava tako što dvaput kliknete na element koga želite da menjete, a zatim odredite sve parametre iz dijaloga. Tekst se dodaje u okviru grafika tako što, kada je aktivan **Chart** prozer, otkucate taj tekst, a zatim ga mišem prevučete na željeno mesto Kod ovog programa da punog uvažaja dolazi moć rada pod *Windows*-om, gde se svi elementi izgleda mogu podesiti samo klikom miša. Na primer, želite da istaknete jednu stavku nekog grafika. Startovaćete grafik torte u tri dimenzije, a sa tim klikvati na mečak torte koji želite da istaknete Oko mečka se pojavljuju tačke koje ga uokviruju Sa da pomeranjem miša IZVUČETE taj deo u odnosu na ceo oko koje su složem ostali delovi. Dobijete grafik gde su svi delovi grafikone u obliku torte složeni normalno – a samo jedan je izvučen i izdvojen u odnosu na ostala. Kao krunu svega možete dodati objašnjenje tog dela grafika, i strelicu od objašnjenja do tog mečka. Kada ste zadovoljni izgledom grafika, u tekst se vraćate preko opcije **Exit and return to ..DOC.**

EDITOR NATPISA



Editor natpisa

Editor natpisa (ili umetnost reči – *WordArt*) se startuje preko **Insert/Object/WordArt** menija ili klikom na ikonu na *toolbar*u. ukoliko ste ovu opciju instalirali Ovaj program, najednostavnije rešene, liči na *Banner* - veoma poznat program za pravljenje natpisa pod DOS-om. Rad sa programom je izuzetno jednostavan, jer se u petpunosti odvija u jednom dijalogu U gornjem delu dijaloga se nalazi mesto za unošenje teksta. Tekst može da zauzima više redova, ali natpisi koje ćete raditi ovim programom su obično kratki i sadrže nekoliko reči. U font delu dijaloga birate jedan od ponuđenih fontova, koji su specifični za ovaj program i ne pojavljuju se u drugim programima. S druge strane, ovaj program ne poznaje njegan drugi font i samim tim NEMA naša slova. Ako vam ta činjenica ne smeta, imate šanse da napravite sasvim uspele natpise. **Size** određuje veličinu slova – **best fit** je poseban slučaj, kada program sam određuje veličinu koja se najbolje uklapa u mesto na papiru. **Style** određuje način ispisa efekat ispisa. Postoji ograničena količina efekata za modifikovanje ispisa. **Fill** određuje način popunjavanja fonta. **Align** uravnava ispis po nekoj od margina ili ga centrira.

Options sadrži još nekoliko sitnijih modifikacija. Ukoliko ste zadovoljni svojim natpisom, možete reći se OK. i natpis je prebačen u tekst. Natpis je moguće ubaciti u bilo kap deo dokumenta, pa i u header ili futer, što može dosta efektno da izgleda.

TOOLBAR

Ispod glavnog menija se nalazi linija se ikonama ili *toolbar* Svaka ikona predstavlja neku od često korišćenih operacija, a osnovna ideja uvođenja *toolbar* linija je da se rad ubrzava kada se često korišćenim opcijama pristupa jednim klikom miša ne preko menija. I stvarno – sve je tu i otvaranje novog dokumenta i štampanje i još puno korisnih stvari. Kako se *toolbar* podešava po vašim željama, možete pročitati u poglavlju „podešavanje Worda", ali da biste znali šta vam nije potrebno, prve morate naučiti čemu koja ikone služi Ikone čemo navoditi počev od leva strane *toolbar*-a.

Otvara novi dokument i počinje rad se „normal" templejtom.

Poziva **Open** opciju **File** menija – dakle, dobijate dijalog se izbor dokumenta ili templeta.

Poziva **Save** opciju **File** menija. Da vas podsetimo da je klik na *toolbar* kratak pestupak, ali je prinsak na **Shift+F12** verovatno još brži.

Poziva **Cut** opciju **Edit** menija. Označeni tekst će biti prebačen u *Clipboard* i istov remeno izbrisan iz teksta.

Poziva **Copy** opciju **Edit** menija. Ova opcija takođe prebacuje blok teksta u *Clipboard*, ali pri tome dokument ostaje sačuvan

Poziva **Paste** opciju **Edit** menija. Sadržaj *Clipboarda* se pranom u tekst na meste kursora.

Ova ikona zamenjuje **Undo** opciju **Edit** menja. **Undo** je brže izvesti pritiskom na **Ctrl+Z.**

Kreira „brojčanu listu" To je način rada koji ćemo koristiti kada navodimo veći broj linija po stavkama. Zamenjuje deo opcije **Bullets and Numbering** iz **Tools** menja.

Ova ikona ima funkciju sličnu prethodnoj, osim ste se sada pri navođenju stavki ne koriste brojevi, nego tačke (crtice)

Ova ikone pomera levu marginu (važi lokalno, za pasus ili za blok teksta) u desnu stranu, skačući za pojedan tab step.

Vrši posao suprotan desetoj – pomera levu marginu ulevo. Ograničenje je fizička leva margina papira.

Ova ikona služi za umetanje tablica u tekst. U gajnom interaktivnom radu se bira veličina tablice: kliknete na ovu ikonu i jednostavne razvučete tablicu na broj polja koji vam je potreban

Ova ikona služi za slaganje teksta u više stubaca (kao novinski tekst. Takođe interaktivan rad i lakoća slaganja teksta.

Ova ikona služi za ubacivanje okvira u tekst. Ukoliko nste u **Page View** modu, program će vam ponuditi prebacivanje u njega rada veće preciznosti postavljanje okvira.

Poziva *Microsoft Draw* program

Poziva *Microsoft Graph* – gajan program za pravljenje grafika.

Ova ikone poziva opciju pravljenja koverta – **Create Envelope** opcija **Tools** menija.

Startuje proveru spelovanja.

Poziva Print opciju File menija.

Ova ikona određuje prikaz „papira" u prozoru, tako da cela papir stane na ekran

Ova ikona prikazuje „papir" u obliku kakav smo i navikli da gledamo – prikaz je sasvim normalne veličine. Koristite je posle prethodne – „skoknuli" ste na momenat u pregled izgleda i želite da se vratite nazad.

Ova ikona startuje prikaz koji je hibrid prethodna dva – vidi se cela papir po horizontali ali ne i po vertikali. Sva slova su još uvek čitljiva, pa se može pisati i u ovom modu

Ispod *toolbar* linije se nalazi još jedna linija sa elementima koji omogućuju formatiranje dokumenta (u *Word* terminologiji ova linija se zove *Ribbon*) Njeni elementi su sledeći.

Takozvani kombo-boks element *Windows*-a. Služi za izbor stila, koji biva primenjen ili na izabrani blok teksta ili na tekst koga nadalje kucamo. Ukoliko kliknemo na taster koji na sebi ima strelicu nadole (pored teksta elementa) kombo-boks se „razmota" i dozvoljava nam da izaberemo stil mišem. Ukoliko kliknemo na tekstualni deo dozvoljeno nam je unošenje naziva stila koga želimo da startujemo, i tu mogućnost ćemo koristiti ukoliko znamo napamet ka-

ke se zove željeni stil. Ovaj element koristimo (osim klika mišem) i uz pomoć tastature – prečica je **Ctrl+S**

Još jedan kombo. Njime biramo font koji ćemo koristiti nadalje u tekstu. Važe sve operacije kao i za prethodnim kombo-boksom, a se tastature se poziva se **Ctrl+F**

U ovom kombo-boksu određujemo veličinu fonta. Ukoliko imamo *True Type* (vektorske) fontove, možemo uneti bilo koju veličinu od 4 do 127 tačaka. Ukoliko nemamo, najbolje je uzeti neku od veličina koje nudi program.

Taster sa slovom **B** uključuje bold (podebljana) slova. Sa tastature se selektuje pritiskom na **Ctrl+B**.

Taster se slovom **I** uključuje italic (iskošena) slova. Prečica je **Ctrl+I**.

Taster se slovom **U** uključuje podvučena slova. Prečica je **Ctrl+U**

Ova četiri tastera služe za uravnavanje pasusa i redom služe za. pasus uravnat po levoj margini (**Ctrl+L**), centriran pasus (**Ctrl+E**) pasus uravnat po desnoj margini (**Ctrl+R**) i pasus uravnat po obe margine (**Ctrl+J**)

Određuju način centriranja tabova.

Prikazuje skrivene karaktere u tekstu – krajeve pasusa, tab karaktere i slično

INDEKS SVIH OPERACIJA SA TASTATURE

Formatiranje teksta

CTRL+F	Promena fonta (pozicionira pointer na font-boks za izbor fonta) Ako još jednom pritisnete **CTRL+F** dobijate izbor karakter stila.
CTRL+P	Promena veličine fonta.
CTRL+F2	Povećava veličinu fonta (svaki put kada pritisnete **CTRL+F2**, dobijate sledeću raspoloživu veličinu fonta)
CTRL+SHIFT+F2	Smanjuje veličinu fonta (svaki put kada pritisnete, dobijate sledeću raspoloživu manju veličinu fonta)
SHIFT+F3	Promena slova u bloku (prvi pritisak na **SHIFT+F3** će sva slova u bloku učiniti velikim, drugi pritisak malim, a treći će sva prva slova u reči napraviti velikima a ostala malima)
CTRL+SPACE	Uklanja dodatno formatiranje iz bloka teksta. Ne može da ukloni formate zadate stilovima, ali sve dodatne može.
CTRL+A	Menja blok teksta tako da cela bude ispisan velikim ili cela malim slovima.
CTRL+B	Naizmenično uključuje i isključuje podebljana slova. Ukoliko smo izabrali blok, umesto se odnosi na njega.
CTRL+D	Dvostruko podvlači tekst. Takođe naizmenično uključuje i isključuje funkciju
CTRL+H	Skriven tekst.
CTRL+I	Iskošena slova, ponašaju se kao i podebljano
CTRL+R	Uključuje i isključuje mala prva slova.
CTRL+U	Podvlači tekst.
CTRL+W	Podvlači reč.

CTRL+=(znak jednakosti)	*Subscript* (tekst koji se piše 3 tačke ispod osnovne linije ispisa)
CTRL+SHIFT+=	*Superscript* (tekst koji se ispisuje 3 tačke iznad osnovne linije)

Da biste formatirali specijalne karaktere, prikažite ih pritiskom na **CTRL+SHIFT+***

Brisanje i umetanje teksta i grafike

BACKSPACE	Briše jedan karakter se leve strane kursora ili izabrani blok teksta.
CTRL+BACKSPACE	Briše jednu reč sa leve strane kursora.
DEL	Briše jednu reč se desne strane kursora ili izabrani blok teksta.
CTRL+DEL	Briše jednu reč se desne strane kursora.
CTRL+X ili SHIFT+DEL	Briše izabrani blok teksta i šalje ga u *Clipboard*
CTRL+Z	Poništava poslednju akciju (undo)
CTRL+F3	Iseca tekst ili grafiku u posebnu vrstu rečnika zvanu „Spike"
CTRL+SHIFT+F3	Umetanje sadržaja *Spike*-a u tekst.
CTRL+C	Kopira tekst ili grafiku u *Clipboard*
CTRL+V or SHIFT+INS	Umeće sadržaj *Clipboarda* u tekst na mesto kursora.
CTRL+F2	Umetanje polja u tekst.

Funkcijski tasteri

F1	Pomoć za izbranu komandu, otvoreni dijalog-boks ili poruku.
ALT+F1	Skok na sledeće polje.
SHIFT+F1	Pomoć za komandu, deo ekrana ili kombinaciju tastera. Kada pritisnete SHIFT+F1, kursor se menja iz strelice u znak pitanja. Kada takvim kursorom kliknete na neko polja na ekranu, dobijate help za njega.
ALT+SHIFT+F1	Skok na prethodno polje.
F2	Kopiranje izabranog bloka teksta.
ALT+F2	Prečica za **Save As** komandu **File** menija.
CTRL+F2	Povećanje veličine fonta na sledeću raspoloživu veličinu.
SHIFT+F2	Pomeranje izabranog bloka teksta.
ALT+SHIFT+F2	Snimanje teksta
CTRL+SHIFT+F2	Smanjuje veličinu fonta na sledeću raspoloživu veličinu.
F3	Umeće u tekst reč iz rečnika
CTRL+F3	Iseca tekst ili grafiku u posebnu vrstu rečnika zvanu „Spike"
CTRL+SHIFT+F3	Umetanje sadržaja Spike-a u tekst.
F4	Ponavlja prethodnu komandu.
ALT+F4	Kraj rada se *Wordom for Windows*
CTRL+F4	Zatvara aktivni dokument.
SHIFT+F4	Ponavlja traženje ili goto komandu
F5	Skok na određenu stranu
CTRL+F5	Restaurira veličinu prozora na prethodnu
SHIFT+F5	Skok na prethodnu poziciju
CTRL+SHIFT+F2	Umetanje dela podstranica.
ALT+F6 ili CTRL+F6	Skok na sledeći dokument ili prozor
ALT+SHIFT+F6 ili CTRL+SHIFT+F2	Skok na prethodni dokument ili prozor
F7	Provera spelovanja izabranog teksta.

SHIFT+F7	Poziv rečnika sinonima.
CTRL+SHIFT+F7	Provera informacija koje su objektno povezane sa dokumentom. Jedna vrsta **makro** komande
F8	Specijalni mod obeležavanja teksta. Jedan pritisak na F8 obeležava jednu reč, dva pritiska obeležavaju pasus. sve do celog teksta.
CTRL+F8	Promena veličine prozora dokumenta.
SHIFT+F8	Sužavanje izbora, opcija suprotna opciji F8.
CTRL+SHIFT+F8	Izbor bloka teksta, ali u modu kolone
F9	Provera polja - proverava validnost polja i menja ga ako je potrebno
ALT+F9	Smanjivanje aplikacije na ikonu
CTRL+F9	Umočenje polja, upotrebom ove opcije kursor se nađe u vitičastim zagradama, a na vama je da unesete polje
SHIFT+F9	Naizmenično uključuje prikazivanje polja ili njihovih rezultata.
ALT+SHIFT+F9	Simulira duplo klik mišem na polje, a služi da se sa tastature koristi gotu polja i polja sa makro naredbama.
CTRL+SHIFT+F9	Poništava promenu polja.
F10	Poziva glavni meni.
ALT+F10	Povećava prozor aplikacije na maksimalnu.
CTRL+F10	Povećava prozor u kome menjamo dokument na ceo ekran.
SHIFT+F10	Rotira izbor sa tastature kroz *annotations, outline, headerfooter*, ili *macro button* bar
CTRL+SHIFT+F10	Aktivira lenjir i njegove ivice sa tastature.
F11	Skok na sledeće polje
CTRL+F11	Zaključavanje polja.
SHIFT+F11	Skok na prethodno polje.
CTRL+SHIFT+F11	Otključavanje polja.
F12	**Save As** opcija **File** menija.
CTRL+F12	**Open** opcija **File** menija.
SHIFT+F12	**Save** opcija **File** menija.
CTRL+SHIFT+F12	**Print** opcija **File** menija.

Kretanje kroz dokument

Strelica levo	Pomera kursor jedan karakter ulevo.
Strelica desno	Pomera kursor jedan karakter udesno
Strelica gore	Pomera kursor jednu liniju nagore.
Strelica dole	Pomera kursor jednu liniju nadole
CTRL+Strelica levo	Pomera kursor za jednu reč ulevo.
CTRL+Strelica desno	Pomera kursor za jednu reč udesno
END	Pomera kursor na početak linije.
HOME	Pomera kursor na kraj linije.
CTRL+Strelica gore	Pomera kursor za jedan pasus nagore
CTRL+Strelica dole	Pomera kursor za jedan pasus nadole
PageDown	Pomera kursor za jedan prozor nadole
PageUp	Pomera kursor za jedan prozor nagore.
CTRL+PageDown	Pomera kursor na dno prozora.
CTRL+PageUp	Pomera kursor na vrh prozora.
CTRL+END	Skok na kraj dokumenta.

CTRL+HOME	Skok na početak dokumenta.

Tasteri za outline mod

ALT+SHIFT+5	Prikaz osnovnog teksta na ekranu u **outline** modu
ALT+SHIFT+F	Naizmenično prikazivanje prve osnovne linije teksta ili celog teksta.
ALT+SHIFT+1	Prikazuje prvi nivo naslova.
ALT+SHIFT+n	Prikazuje n-ti nivo naslova. Na primer, da bi prikazali nivo 5 pritisnućemo **ALT+SHIFT+5.**
ALT+SHIFT+9	Prikazuje deveti nivo naslova.
ALT+SHIFT+A	Prikazuje sav tekst ili sve naslove ako je ceo tekst već vidljiv

Formatiranje pasusa

CTRL+1	Formatira linije sa jednim redom proreda.
CTRL+2	Formatira linije sa dva reda proreda.
CTRL+5	Formatira linije sa jednim i po redom proreda.
CTRL+O (slovo O)	Dodaje jednu praznu liniju ispred pasusa.
CTRL+0 (cifra 0)	Briše praznu liniju ispred pasusa.
CTRL+E	Centrira pasus.
CTRL+J	Formira pasus uravnat po obe ivice.
CTRL+L	Formira pasus uravnat po levoj ivici.
CTRL+R	Formira pasus uravnat po desnoj ivici.
CTRL+N	Uvlači ceo pasus od leve margine za jedan tab.
CTRL+M	Uvlači ceo pasus ka levoj margini za jedan tab
CTRL+T	Pravi „viseći", za jedan tab uvučeni pasus.
CTRL+G	Za jedan tab izvlači ka levoj margini „viseći" pasus.
CTRL+S	Dodaje stil pasusu
ALT+SHIFT+5 (na numeričkoj tastaturi)	Dodaje *Normal* stil pasusu.

Obeležavanje teksta (select)

Bilo koji tekst se obeležava tako što dovedemo kursor na jedan kraj budućeg bloka, pritisnemo i držimo SHIFT, a zatim kursor uz pomoć tastera sa strelicama pomerimo do drugog kraja bloka. Tek tada pustimo SHIFT taster. Takođe tekst se može odabrati korišćenjem **Extend** (F8) tastera.

SHIFT+Strelica desno	označava jedan karakter desno od kursora.
SHIFT+Strelica levo	označava jedan karakter levo od kursora.
CTRL+SHIFT+Strelica desno	označava tekst od kursora do desnog kraja reči
CTRL+SHIFT+Strelica levo	označava tekst od kursora do levog kraja reči
SHIFT+END	označava tekst od kursora do kraja linije.
SHIFT+HOME	označava tekst od kursora do početka linije.
SHIFT+Strelica dole	označava jednu liniju nadole.
SHIFT+Strelica gore	označava jednu liniju nagore
CTRL+SHIFT+Strelica desno	označava tekst od kursora do kraja pasusa.
CTRL+SHIFT+Strelica levo	označava tekst od kursora do početka pasusa.
SHIFT+PageDown	označava jedan ekran teksta od kursura ka kraju teksta.
SHIFT+PageUp	označava jedan ekran teksta od kursura ka početku teksta.
CTRL+SHIFT+END	označava tekst od kursora do kraja dokumenta.
CTRL+SHIFT+HOME	označava tekst od početka dokumenta do kursora.
CTRL+5 (sa num. tastature)	selektuje ceo dokument
CTRL+SHIFT+F8	selektuje kolonu teksta. Posle toga koristite tastere sa strelicama. Za kraj biranja pritisnite ESC.

F8	takozvani **Extend** taster, koji vam omogućava da na specijalan način obeležite tekst.

- Dovedite kursor na jedan ugao teksta koga želite da obeležite
- Pritisnite **F8**. Natpis **EXT** se pojavljuje na statusnoj brajli (najniže linije na ekranu)
- Sada birate tekst na neki od više mogućih načina. Ukoliko pritiskate **F8** više puta količina teksta koje je izabrana će rasti po pravilu reč, rečenica, pasus, dokument. Da biste smanjili količinu teksta na prethodno odabranu, pritisnite **SHIFT+F8**. Koristeći tastere sa strelicama, taste **PageDown, PageUp, Home,** i **End** birate tekst koji želite. U ovom modu možete pritisnuti bilo koji karakter na primer tačku (.), **TAB** ili **ENTER** da bi tekst bio izabran do tog karaktera.
- Za izlazak iz **Extend** moda pritisnite **ESC**

Lenjir

CTRL+SHIFT+F10	Aktivira lenjir
HOME	Pomeranje kursora na levu ivicu lenjira.
END	Pomeranje kursora na desnu ivicu lenjira.
L	Podešava položaj mesta levog uvlačenja pasusa.
R	Podešava položaj mesta desnog uvlačenja pasusa.
F	Podešava uvlačenje prve linije pasusa.
INS	Postavlja tab stop na lenjiru.
DEL	Briše sva tab stop mesta.
ENTER	Potvrđivanje svih podešavanja i povratak u unos teksta.
ESC	Izlaz iz podešavanja bez promena i povratak u unos teksta.

Rad sa tabelom

Kada kucate tekst u tabeli, on se unosi u okviru jedne ćelije. Kada dođete do poslednje ćelije u tabeli, možete je proširiti za još jedan red pritiskom na **TAB** taster

ENTER	Unosi novi pasus u okviru ćelije
TAB	Prebacuje kursor u sledeću ćeliju tabele.
SHIFT+TAB	Prebacuje kursor u prethodnu ćeliju tabele
ALT+HOME	Prebacuje kursor u prvu ćeliju reda u kome je kursor
ALT+END	Prebacuje kursor u poslednju ćeliju reda u kome je kursor
ALT+PAGE UP	Prebacuje kursor u prvu ćeliju kolone u kojoj je kursor
ALT+PAGE DOWN	Prebacuje kursor u poslednju ćeliju kolona u kojoj je kursor
Strelica gore	Pomera kursor u prethodni red.
Strelica dole	Pomera kursor u sledeći red.
CTRL+TAB	Umeće TAB karakter u ćeliju

Rad u poljima

ALT+SHIFT+D	Unošenje datumskog polja.
ALT+SHIFT+F	Unos *merge* polja (važi samo u glavnom dokumentu)
ALT+SHIFT+F	Unos polje koje prikazuje broj strane.
ALT+SHIFT+T	Unos polje koje prikazuje vreme
CTRL+SHIFT+F7	Potvrđivanje usklađenosti informacija u objektno povezanim poljima.
F9	Provera polja, opcija proverava validnost polja i menja ga ako je to potrebno
CTRL+F9	Unošenje polje upotrebom opcije kursor sa nada u vitičastim zagradama, a na vama je da unesete polje
SHIFT+F9	Naizmenično uključuje prikazivanje polja ili njihovih resultata.
ALT+SHIFT+F9	Simulira dupli klik mišem na polje, a služi tome de sa sa tastature koriste **goto** polja i polja sa makro naredbama.
CTRL+SHIFT+F9	Poništava promenu polja.

F11	Skok na sledeće polje.
CTRL+F11	Zaključavanje polja.
SHIFT+F11	Skok na prethodno polje.
CTRL+SHIFT+F11	Otključavanje polja.

Prečice pri radu sa mišem

„Klikom" miša na određene oblasti ekrana možete „prečicom" doći do mnogih operacija.

Dvostruki klik na bilo koji tab stop otvara dijalog-boks za podešavanje tabova.

Dvostruki klik bilo gde na gornjoj polovini lenjira otvara dijalog za formatiranje pasusa.

Dvostruki klik na statusnoj liniji (donji levi ugao prozora) otvara **Goto** dijalog za brzi skok u neki drugi deo dokumenta.

Dvostruki klik bilo gde na *toolbar* liniju osim na neku ikonu otvara **Options** dijalog koji nam služi za podešavanje *Worda*

Dvostruki klik na broj fusnote u tekstu otvara spisak fusnota.

Dvostruki klik na liniju ispod *toolbar*-a, osim na ikonu ili na polja za unos daje dijalog boks za podešavanje parametara karaktera.

Dvostruki klik na print merge liniju daje **print merge** dijalog boks.

Dvostruki klik na bilo koji simbol unesen pomoću symbol opcije insert menija poziva **symbol** opciju

Dvostruki klik na naslovnu liniju *Worda* na vrh ekrana vraća opcije **Maximize** i **Restore.**

Dvostruki klik na anotaciju daje spisak anotacija.

Dvostruki klik na bilo koji objektno povezani deo dokumenta (proizvod nekog od spoljnih programa) poziva sam taj program i istovremeno učitava taj objekt. Na primer dvostruki klik na formulu poziva **equation editor** i učitava formulu u njega.

Obeležavanje teksta mišem

Jednu reč označavamo tako što dvaput kliknemo na nju.

Jednu liniju označavamo tako što dovedemo strelicu miša sa leve strane linije, na samu levu ivicu prozora, a zatim kliknemo jednom.

Više linija teksta označavamo tako što ponovimo postupak sa jednu liniju, sa tom izmenom da sada držimo taster miša , prevučemo" ga pored tih linija, a tak onda pustimo

Jednu rečenicu označavamo tako što pritisnamo **Ctrl** taster i kliknemo bilo gde u okviru te linije.

Pasus označavamo isto kao i jednu liniju teksta, s tom razlikom što je sada potrebno dva puta kliknuti.

Više paragrafa biramo duplim klikom na levoj ivici ekrana, a zatim povlačenjem miša po ivici

Ceo dokument sa lakoća tako što pritisnemo **Ctrl** taster i kliknemo mišem na levoj ivici prozora.

Jednu kolonu teksta biramo tako što DESNIM tasterom na mišu kliknemo na jedan kraj bloka, i razvučemo selektor na ceo blok.